UNIVERSIDADE DA CORUÑA

E. T. S. DE INGENIEROS DE
CAMINOS, CANALES Y PUERTOS

PhD THESIS

HIGH-ORDER FINITE VOLUME METHODS BASED ON MOVING LEAST SQUARES FOR COMPUTATIONAL FLUID DYNAMICS. APPLICATION TO ALL-SPEED AND INCOMPRESSIBLE FLOWS ON UNSTRUCTURED GRIDS

BY

LUIS RAMÍREZ PALACIOS

ADVISORS:

XESÚS NOGUEIRA GAREA
IGNASI COLOMINAS EZPONDA

A CORUÑA - MARCH 2015

To my family.

Contents

List of Figures **v**

List of Tables **xiii**

1 Introduction. **1**
1.1 Introduction . . . 1
1.2 Motivation . . . 5
1.2.1 Impact of high-order methods. . . . 9
1.2.2 High-order methods for turbomachinery problems . . . 17
1.2.3 State of art . . . 21
1.3 Objectives . . . 29
1.4 Overview of the thesis . . . 30

2 Finite Volume, and Moving Least Squares approximants. The FV-MLS method. **31**
2.1 Introduction . . . 31
2.2 The Finite Volume Method . . . 32
2.3 The Moving Least Squares method . . . 37
2.3.1 Computational derivatives . . . 40
2.3.2 The smoothing function or *kernel* . . . 41
2.4 The FV-MLS method . . . 43
2.5 Conclusions . . . 46

3 A high-order FV-MLS formulation for compressible flows at all-speed. **49**
3.1 Introduction . . . 49
3.2 General formulation . . . 51
3.2.1 Time integration . . . 51
3.2.2 Numerical Fluxes . . . 52
3.2.3 Approximate Riemann solvers . . . 52
3.2.4 Low Mach entropy fixes . . . 54
3.3 Obtaining physical solution using higher-order MLS reconstructions . . . 55
3.3.1 Accuracy assessment and pressure scaling for low-Mach computations 56

3.4 Low-Mach fixes and limiters . . . 64
3.4.1 Slope limiters . . . 65
3.4.2 MLS-based shock sensor . . . 69
3.4.3 Roe scheme with Rieper's fix . . . 70
3.4.4 Rusanov scheme with Li and Gu's fix . . . 72
3.5 3D decay of compressible isotropic turbulence . . . 72
3.6 Unsteady transonic viscous flow over a circular cylinder . . . 74
3.7 Conclusions . . . 77

4 A high-order FV-MLS formulation for incompressible flows. 81
4.1 Introduction . . . 81
4.2 Governing equations . . . 83
4.2.1 Pressure based methods . . . 84
4.2.2 Artificial Compressibility Method . . . 85
4.2.3 Methods Based on Derived Variables . . . 86
4.3 Numerical discretization of the incompressible Navier-Stokes equations . . . 86
4.3.1 Momentum equation . . . 87
4.3.2 Velocity and pressure correction . . . 92
4.4 Numerical Results . . . 99
4.4.1 Kovasznay Flow . . . 99
4.4.2 2D Taylor-Green Flow . . . 102
4.4.3 Lid-driven cavity Flow . . . 105
4.4.4 Laminar Flow around a Cylinder . . . 109
4.5 Conclusions . . . 112

5 Moving grids. Development of high-order sliding-mesh techniques. 115
5.1 Introduction . . . 115
5.2 Governing equations and numerical methods . . . 117
5.3 General formulation . . . 118
5.4 MLS-based sliding mesh . . . 119
5.4.1 MLS-based sliding mesh with intersections . . . 120
5.4.2 Interface-halo-cell sliding mesh . . . 122
5.5 Numerical Examples . . . 127
5.5.1 Ringleb flow . . . 128
5.5.2 Vortex Convection . . . 133
5.5.3 Supersonic Flow over a cylinder . . . 135
5.5.4 Cross-flow turbines . . . 139
5.6 Conclusions . . . 156

6 Conclusions and further researh 157
6.1 Further research . . . 159

Bibliography 161

List of Figures

1.1 Richardson's map grid. From [175]. . . . 3
1.2 Richardson's forecast-factory. From [128]. . . . 3
1.3 Classification of the flow as function of the Mach number. The limits are taken from [59]. . . . 4
1.4 Schematic representation of the flow patterns around a cylinder for different Reynolds number. From [84]. . . . 5
1.5 Different types of grid: a) Structured mesh and b) Unstructured mesh. From [145]. . . . 6
1.6 Schematic representation of the Ringleb's flow. From [130]. . . . 7
1.7 Numerical error for each mesh size with different orders of accuracy. Data taken from [39]. . . . 8
1.8 Influence of the order of resolution in the one-dimensional linear wave problem at $t = 400$. . . . 10
1.9 Diagram of the forces exerted on the NACA 0012. . . . 11
1.10 Close view of the unstructured meshes employed for the computation of the inviscid flow past a NACA0012 test case. Mesh A is the coarsest unstructured mesh with 5322 quadrilateral elements. The finest mesh is denoted by mesh B and has 12246 quadrilateral elements. . . . 12
1.11 Mach field and contours around the NACA 0012 profile for different orders of accuracy. . . . 12
1.12 Entropy field and contours around the NACA 0012. The maximum increment of entropy, dS_{max}, is indicated for each case. . . . 14
1.13 Schematic representation of the flow structure around a cylinder for a Reynolds number $5 < Re < 200$. From [84]. . . . 15
1.14 Close detail of the mesh around a cylinder . . . 15
1.15 Comparison of the vorticity around the cylinder for different discretization orders and time. . . . 16
1.16 Time history of the lift coefficient for a low-order scheme, a), and high-order, b). . . . 17
1.17 Different turbines for renewable power generation [165]. Dashed lines denote the motion region. . . . 18

1.18 Different tidal and wind turbines, a) Horizontal Axis Wind Turbine (HAWT) [212], b) Vertical Axis Wind Turbines (VAWT) [209], and c) Tidal Turbine [155]. 19
1.19 Aerial view of the Horns Rev wind farm in Denmark. From [50]. 20

2.1 Control volume definitions for finite volume methods: a) cell-centered and b) vertex-centered approach. 32
2.2 Definition of the control volume Ω_I. 33
2.3 Schematic illustration of the cell I and its integration points ("$\mathbf{x}$"). 34
2.4 Constant piecewise solution. 35
2.5 Piecewise linear reconstruction inside the cells. Notice the discontinuity across the interface j. 36
2.6 Compact support ($\Omega_{\boldsymbol{x}}$) centered in point P. 38
2.7 Influence of the parameter s_x in the exponential *kernel* and its first derivative 42
2.8 Typical stencil for interior cells used for cubic MLS approximation centered at cell centroid I. Shaded cells represent the stencil of cell I 43
2.9 Typical stencil for interior cells used for cubic MLS approximation centered at cell centroid I. k_i represents the neighboring cells of cell I 44
2.10 Stencil for interior cells used for cubic MLS approximation centered at integration point j. k_i represents the neighboring cells of the integration point j 45
2.11 Reconstructed variables used to evaluate the inviscid fluxes at the integration point at the edges of the control volume Ω_I. From [37]. 45
2.12 MLS stencil for boundary cells. Note that ghost cells are included in the stencil of cell I. 47

3.1 Piecewise linear reconstruction. Notice that the increase of the order of the reconstruction reduces the term $\Delta(\boldsymbol{U})$. 54
3.2 Close view of the coarsest and the finest structured O-grids employed for the computation of the inviscid flow past a circular cylinder test case. The coarsest mesh (left) has 32×16 elements and the finest (right) 96×48 elements. 56
3.3 Pressure contours for inviscid flow past a cylinder test case for $M_\infty = 10^{-3}$. The solution is obtained in the 96×48 grids by using a first order Roe scheme (a) and by using the 4^{th} order FV-MLS Roe scheme (b). 57
3.4 Inviscid flow past a cylinder test case. a)Pressure-Mach scaling for the 3^{rd} and 4^{th} order ROE-FV-MLS scheme in the 32×16 grid. b) Influence of the grid on the accuracy problem. Pressure-Mach scaling for the 4^{th} order ROE-FV-MLS scheme using different grids. 59
3.5 Inviscid flow past a cylinder test case. Pressure-Mach scaling for the 4^{th} order ROE-FV-MLS scheme with Rieper's Fix in the 32×16 grid. 61
3.6 Inviscid flow past a cylinder test case. Pressure contours for $M_\infty = 10^{-6}$. The solution is obtained in the 32×16 grid by using the 4^{th} order ROE-FV-MLS scheme and the low-Mach fix of Rieper. 62

3.7 Inviscid flow past a cylinder test case. Mach contours for $M_\infty = 10^{-6}$. The solution is obtained in the 32×16 grid by using the 4^{th} order ROE-FV-MLS scheme and the low-Mach fix of Rieper. 62
3.8 Inviscid flow past a cylinder test case. Pressure contours for $M_\infty = 10^{-2}$. The solution is obtained in the 96×48 grid by using the Li and Gu's low-Mach fix with a first order FV Rusanov scheme (a) and the 4^{th} order RUS-FV-MLS scheme. It is observed that the first order scheme presents a weak checkerboard that is removed with the use of the high-order scheme. 64
3.9 Inviscid flow past a cylinder test case. Comparison of the pressure-Mach scaling for the 4^{th} order RUS-FV-MLS scheme with and without the Li and Gu's Fix for the 32×16 grid. 66
3.10 Reconstructed solution using a) unlimited piecewise linear and b) limited piecewise linear reconstruction. 66
3.11 Close view of the unstructured O-grid employed for the computation of the inviscid flow past a circular cylinder test case using slope limiters. 70
3.12 Inviscid $M_\infty = 10^{-3}$ flow past a cylinder on an unstructured grid. Pressure contours. Rieper's low-Mach fix with non-differentiable limiters: Barth-Jespersen limiter (a), Van Albada limiter (b). Rieper's low-Mach fix with Venkatakrishnan limiter (c) and with Van Albada limiter and MLS-based sensor (d). 71
3.13 Inviscid $M_\infty = 10^{-3}$ flow past a cylinder on an unstructured grid. Pressure contours. Rusanov's low-Mach fix with non-differentiable limiters: Barth-Jespersen limiter (a), Van Albada limiter (b). Rieper's low-Mach fix with Venkatakrishnan limiter (c) and with Van Albada limiter and MLS-based sensor (d). 73
3.14 Time history of the turbulent kinetic energy decay. The solution is obtained with 3^{rd} ROE-FV-MLS with Rieper's Fix on a structured mesh with 32^3 elements. 74
3.15 Instantaneous three-dimensional energy spectrum at $t/\tau_0 = 0.3$. The solution is obtained with 3^{rd} ROE-FV-MLS with Rieper's Fix on a structured mesh with 32^3 elements. 75
3.16 Unsteady transonic viscous flow over a circular cylinder. Magnitude of the temperature gradient near the cylinder for $t^* = 49.93$. The solution is obtained with 3^{rd} ROE-FV-MLS with Rieper's Fix and Van Albada limiter with MLS-based sensor. 76
3.17 Unsteady transonic viscous flow over a circular cylinder. Mach number field for $t^* = 49.93$. The solution is obtained with 3^{rd} ROE-FV-MLS with Rieper's Fix and Van Albada limiter with MLS-based sensor. 77
3.18 Unsteady transonic viscous flow over a circular cylinder. Pressure field and selective limiting for $t^* = 49.93$. Shaded cells indicates the elements where the slope limiter is activated. The solution is obtained with 3^{rd} ROE-FV-MLS with Rieper's Fix and Van Albada limiter with MLS-based sensor. . . 78
3.19 Unsteady transonic viscous flow over a circular cylinder. Surface pressure coefficient around the cylinder. The experimental data is taken from [142]. . 79

4.1 Staggered grid variables arragement. Figure taken from [85] 83
4.2 The unsteady SIMPLE algorithm flowchart 87
4.3 Schematic representation of cell I and its integration point j. 91
4.4 One-dimensional structured grid 93
4.5 Checkerboard pressure field 94
4.6 One dimensional checkerboarded pressure field 95
4.7 Analytical conditions of the Kovasznay flow for $Re = 40$. Figures a) and b) represents the u-velocity and v-velocity fields, and figure c) represents the pressure field and contours. 101
4.8 Initial conditions of the Taylor Green flow for $Re = 100$. Figures a) and b) represents the u-velocity and v-velocity fields, and figure c) represents the pressure field and contours. 103
4.9 L_2 norm of error of velocity and pressure fields as function of the number of cells for the Taylor-Green vortex flow with $Re = 100$. The FV-MLS scheme computations are based on two Gauss points quadrature rules. Dashed line denotes the slope of the expected formal order of accuracy. 105
4.10 a) Lid-driven cavity flow configuration and boundary conditions. b) Unstructured mesh of 1635 quadrilateral cells used for the driven cavity flow problem. 106
4.11 Isocontours of the u velocity field a), the v velocity field, b), the pressure field c) and streamlines d). Numerical solution of the Cavity flow at $Re = 1000$ and computed using the fourth-order FV-MLS scheme. 107
4.12 Comparison of the velocity profiles for different numerical orders of accuracy obtained on a 1635 control volumes unstructured mesh at $Re = 1000$. The numerical solutions were obtained with the FV-MLS method and the commercial software ANSYS/FLUENT 14.0.(*) The solution is obtained with third-order in the convective velocity and second-order in pressure. 108
4.13 Velocity profiles at different Reynolds number obtained on a 1635 control volumes unstructured mesh using the fourth-order FV-MLS scheme. 109
4.14 Geometry description for the laminar flow around a cylinder 110
4.15 Unstructured meshes used for the FV-MLS computations of the Laminar Flow around a Cylinder. The coarsest mesh (a) is denoted by mesh A and has 4968 quadrilateral elements and the finest mesh (b) is denoted by mesh B and has 19079 quadrilateral elements. 111
4.16 Velocity field and streamline pattern of the steady closed wake around a cylinder at $Re = 20$ obtained using third order FV-MLS scheme on the finer mesh. 112
4.17 Pressure field and contours around a cylinder at $Re = 100$ obtained using third-order FV-MLS scheme 113
4.18 Vorticity field around the cylinder at $Re = 100$ obtained using third-order FV-MLS scheme 113

5.1 Schematic illustration of the sliding mesh concept. The top zone slides over the fixed grid. Note the non-conformal grid. 119
5.2 Schematic illustration of stencil for MLS interpolation at interface. 120

5.3 Schematic representation of intersection nodes. 121
5.4 Schematic representation of the flux exchange for the MLS-based sliding mesh with intersections approach. 121
5.5 Schematic representation of the Full Stencil approach. 122
5.6 Schematic representation of the Half Stencil approach. 123
5.7 Schematic representation of Interface-halo-cell sliding mesh. Note that the flux can be computed without the definition of intersections 123
5.8 Schematic representation of Interface-halo-cell sliding mesh with full stencil 124
5.9 Schematic view of the computational domains at $x = 5.0$ with the corresponding halo cell (dashed line) for each domain at the interface. 125
5.10 a) Time evolution of the numerical results of the 1D steady shock when the FS Halo cell approach is employed with $s_x = 4.5$. The interface is located at $x = 5$. A numerical oscillation propagation can be observed. After 160 steps the stationary state is reached but the shock is not preserved. b)Comparison of the numerical results of the 1D steady shock between the FS Halo cell approach with $s_x = 6.0$ and when a single mesh grid is employed to discretize the domain. No discrepancies are observed. 126
5.11 Numerical solutions at $t = 0.2$ for the 1D Unsteady Shock with 3^{rd} order FV-MLS method. The interface is located at $x = 0.5$. 127
5.12 a) Geometry description of Ringleb flow problem. b) Unstructured mesh of 580 triangles. Note the non-conformal mesh at the interface 128
5.13 L_2 norm of the entropy error as function of the number of cells for the Ringleb flow with $\omega = 0$ rad/s. The FV-MLS scheme computations are based on three Gauss points quadrature rules. Dashed line denotes the slope of the expected formal order of accuracy. 131
5.14 L_2 norm of the conservation error as function of the number of cells for the Ringleb flow for $\omega = 0$ rad/s and $\omega = 0.01$ rad/s. The FV-MLS scheme computations are based on three Gauss points quadrature rules. Dashed line denotes the slope of the expected formal order of accuracy. 132
5.15 a) Geometry description of Two Dimensional Vortex Convection problem. b) Unstructured mesh of 3884 quadrilateral elements. 134
5.16 Geometry description of the Supersonic Flow over a cylinder of radius $R1 = 1$. The shaded ring, with inner radius $R2 = 2$ and outer radius of $R3 = 3$, denotes the rotational zone. 136
5.17 Structured non-conformal mesh of 7200 quadrilateral elements and closer view to the cylinder and the interfaces. 136
5.18 Schematic representation of the stand-off distance. 137
5.19 Comparison of Mach field for the supersonic flow over a cylinder. The solutions are obtained using third-order FV-MLS scheme. Dashed line denotes the solution obtained with a single mesh, purple line is obtained with FS Intersections approach and blue line refers to the solution obtained when the FS Halo approach is employed. The interfaces are highlighted in red. 138
5.20 Comparison of the C_p distribution the supersonic flow over a cylinder. The solutions are obtained using third-order FV-MLS scheme. 138

5.21 Different types of cross-flow turbines. From [67] 139
5.22 Diagram of velocities, angles and frames of references on a single blade. The cartesian frame of reference is represented by (x, y), the drag-lift fram of reference by (D, L) and the normal-tangential frame by (N, T) 140
5.23 Representation of the angle of attack (α) as function of the angular location (θ) for tip speed ratio of $\lambda = 2$. 141
5.24 Schematic description of the one-bladed cross-flow turbine 142
5.25 Close view of the unstructured mesh around the single-bladed cross flow turbine. The red line denotes the interface 142
5.26 a) Normalized normal force and b) normalized tangential force against azimuth for $\lambda = 1, 2, 5$ for the one-bladed cross-flow turbine. 143
5.27 Velocity contours around the one bladed cross flow turbine. The solution is obtained with a 3^{rd} FV-MLS method after six rotation cycles ($\theta = 1800$ deg) with $\lambda = 5$ 144
5.28 Flow around the one bladed cross-flow turbine, vorticity field (a) and pressure field (b).The solution is obtained with a 3^{rd} FV-MLS method after six rotation cycles ($\theta = 1800$ deg) with $\lambda = 5$ 145
5.29 Schematic description of the three-bladed cross-flow turbine. 146
5.30 Close view of the unstructured mesh around the cross flow turbine at time $t = 5.0$. Note the non-conformal mesh at the interface. The red line denotes the interface 146
5.31 Flow around the three bladed cross-flow turbine, velocity field (a) and pressure field (b). The solution is obtained with a third-order FV-MLS method at $t = 5.0$. 147
5.32 Vorticity field and contours around the three bladed cross-flow turbine. The solution is obtained with a third-order FV-MLS method at $t = 5.0$. 148
5.33 Normalized tangential and normal forces against azimuth for one of the blades of the three-bladed cross-flow turbine. 149
5.34 Normalized drag and lift forces against azimuth for one of the blades of the three-bladed cross-flow turbine. 149
5.35 Velocity field and contours around the three bladed cross-flow turbine. The solutions are obtained using a third-order FV-MLS method. The solution is plotted for the following angular locations: $\theta = 360, 450, 540, 630, 720, 810$ deg. Note that no numerical artifacts near the interface are observed. . . . 150
5.36 Velocity field and contours around the three bladed cross-flow turbine. The solutions are obtained using a third-order FV-MLS method. The solution is plotted for the following angular locations: $\theta = 900, 990, 1080, 1170, 1260, 1350$ deg. Note that no numerical artifacts near the interface are observed. . . . 151
5.37 Pressure field and contours around the three bladed cross-flow turbine. The solutions are obtained using a third-order FV-MLS method. The solution is plotted for the following angular locations: $\theta = 360, 450, 540, 630, 720, 810$ deg. Note that no numerical artifacts near the interface are observed. . . . 152

5.38 Pressure field and contours around the three bladed cross-flow turbine. The solutions are obtained using a third-order FV-MLS method. The solution is plotted for the following angular locations: $\theta = 900, 990, 1080, 1170, 1260, 1350$ deg. Note that no numerical artifacts near the interface are observed. . . . 153
5.39 Vorticity field around the three bladed cross-flow turbine. The solutions are obtained with a third-order FV-MLS method for the following angular locations: $\theta = 360, 450, 540, 630, 720, 810$ deg. Note that no numerical artifacts near the interface are observed. 154
5.40 Vorticity field around the three bladed cross-flow turbine. The solutions are obtained with a third-order FV-MLS method for the following angular locations: $\theta = 900, 990, 1080, 1170, 1260, 1350$ deg. Note that no numerical artifacts near the interface are observed. 155

List of Tables

1.1 Comparison of the drag and lift coefficients, C_D and C_L, for the inviscid flow around a NACA 0012 for different meshes and orders of accuracy. 13
1.2 Tidal and wind operational fluid speeds for a vertical axis turbine. Data taken from [56]. 19

3.1 Inviscid flow past a cylinder test case. Accuracy orders for the 4^{th} ROE-FV-MLS scheme for different Mach numbers. For $M_\infty = 10^{-3}$ we have only obtained a physical solution for the finest grid. 58
3.2 Inviscid flow past a cylinder test case. Accuracy orders for the 3^{rd} and 4^{th} order ROE-FV-MLS scheme with Rieper's Fix for different Mach numbers. 60
3.3 Inviscid flow past a cylinder test case. Accuracy orders for the 4^{th} order RUS-FV-MLS scheme for different Mach numbers. 63
3.4 Inviscid flow past a cylinder test case. Accuracy orders for the 4^{th} order RUS-FV-MLS scheme with Li and Gu's fix for different Mach numbers. . . 65
3.5 Inviscid $M_\infty = 10^{-3}$ flow past a cylinder on an unstructured grid. Comparison of the drag coefficient for different slope limiters. The solutions are obtained with a fourth order Rieper's Fix Roe FV-MLS scheme on an unstructured mesh. 72
3.6 Unsteady transonic viscous flow over a circular cylinder. Comparison of the drag coefficient with other numerical and experimental results. The 2D ROE-FV-MLS solution is obtained with a 3^{rd} order Roe FV-MLS method with Rieper's Fix. 78

4.1 Accuracy orders of velocity components and pressure field for Kovasznay flow test case. Re=40, cubic MLS and one Gauss point is employed. The order of convergence is stacked at two. 101
4.2 Accuracy orders of velocity components and pressure field for Kovasznay flow test case. Re=40, cubic MLS and two Gauss points are employed. The expected order of convergence is obtained. 102
4.3 Accuracy orders and L_2 of velocity components and pressure field for the Taylor-Green vortex test case at $Re = 100$ and $t = 10^{-2}$ for the third-order FV-MLS scheme. 103

4.4 Accuracy orders and L_∞ of velocity components and pressure field for the Taylor-Green vortex test case at $Re = 100$ and $t = 10^{-2}$ for the third-order FV-MLS scheme. 104
4.5 Accuracy orders and L_2 of velocity components and pressure field for the Taylor-Green vortex test case at $Re = 100$ and $t = 10^{-2}$ for the fourth-order FV-MLS scheme. 104
4.6 Accuracy orders and L_∞ of velocity components and pressure field for the Taylor-Green vortex test case at $Re = 100$ and $t = 10^{-2}$ for the fourth-order FV-MLS scheme. 104
4.7 Comparison of the drag and lift coefficients, C_D and C_L, the increment of pressure Δp and the length of recirculation, L_a for different orders of accuracy and meshes. 111
4.8 Unsteady flow around a cylinder at $Re = 100$. Comparison of the maximum drag and lift coefficients, C_{Dmax} and C_{Lmax}, the Strouhal number and the increment of pressure Δp for different orders of accuracy of the FV-MLS method. 112

5.1 Accuracy orders, conservation error and L_2 norm of entropy error for the Ringleb flow test case for the third-order FV-MLS scheme employing the three different sliding mesh approaches on non-conformal meshes. 129
5.2 Accuracy orders, conservation error and L_2 norm of entropy error for the Ringleb flow test case for the fourth-order FV-MLS scheme employing the three different sliding mesh approaches on non-conformal meshes. 130
5.3 Accuracy orders, conservation error and L_2 norm of entropy error for the Ringleb flow test case for the third-order FV-MLS scheme employing the three different approaches described in this Chapter with an angular velocity of $\omega = 0.01$ rad/s. 131
5.4 Accuracy orders, conservation error and L_2 norm of entropy error for the Ringleb test case for the fourth-order FV-MLS scheme employing the three different approaches described in this Chapter with an angular velocity of $\omega = 0.01$ rad/s. 132
5.5 Accuracy orders, conservation error and L_2 norm of variables error for the two dimensional vortex convection for the third-order FV-MLS scheme employing the three different approaches described in this Chapter with an angular velocity of $\omega = 0.0$ rad/s. 134
5.6 Accuracy orders, conservation error and L_2 norm of variables error for the two dimensional vortex convection for the third-order FV-MLS scheme employing the three different approaches described in this Chapter with an angular velocity of $\omega = 1.0$ rad/s. 135
5.7 Comparison of the normalized stagnation pressure p_0 and the stand-off distance obtained using a single mesh and the three sliding mesh methods (present computations were performed using a 3rd order FV-MLS scheme) . 137
5.8 Initial Conditions for the One-bladed cross flow turbine 143

Nomenclature

- ALE: Arbitrary Lagrangian Eulerian.
- BJ: Barth-Jespersen limiter.
- CAD: Computer Aided Design.
- CENO: Central Essentially Non-Oscillatory.
- CFD: Computational Fluid Dynamics.
- CFL: Courant–Friedrichs–Lewy number.
- DG: Discontinuos Galerkin.
- DLM: Diffuse Element Method.
- DNS: Direct Numerical Solution.
- EFG: Element-Free Galerkin.
- ENO: Essentially Non-Oscillatory.
- FD: Finite Difference Method.
- FEM: Finite Element Method.
- FS: Full Stencil.
- FV: Finite Volume Method.
- FV-MLS: Finite Volume method based on Moving Least Squares method
- GLS: Galerkin Least Squares.
- HAWT: Horizontal Axis Wind Turbines
- HS: Half Stencil.
- IB: Immersed Boundary method.

- IGA: Isogeometric Analysis.
- ILES: Implicit Large-Eddy Simulation.
- LEE: Linearized Euler Equations.
- LES: Large-Eddy Simulation.
- MAC: Maker And Cell.
- MIM: Momentum Interpolation Method.
- MLS: Moving Least Squares.
- MRF: Multiple Reference Frames.
- MUSCL: Monotone Upstream Schemes for Conservation Laws.
- NURBS: Non-Uniform Rational B-Splines.
- OMIM: Original Momentum Interpolation Method.
- PDE: Partial Differential Equation.
- PISO: Pressure Implicit with Splitting Operators.
- PPM: Piecewise Parabolic Method.
- PSPG: Pressure Stabilizing Petrov Galerkin.
- RK: Runge-Kutta. Runge-Kutta method for temporal integration.
- SD: Spectral Difference Method.
- SM: Sliding Mesh method.
- SUPG: Streamline Upwind Petrov Galerkin.
- SV: Spectral Volume Method.
- SIMPLE: Semi Implicit Method for Pressure Linked Equations.
- SIMPLEC: SIMPLE Corrected.
- SIMPLER: SIMPLE Revised.
- TVD: Total Variation Diminishing.
- VAWT: Vertical Axis Wind Turbines
- VMS: Variational Multiscale Method.
- WENO: Weighted Essentially Non-Oscillatory.

Acknowledgements

I have to confess that this acknowledgements section was one of the most difficult parts of the dissertation to write because it is not possible to sufficiently express my deep appreciation to so many people without whom I would not have been able to complete this research.

First of all, I would like to express my sincere appreciation to my advisors: Xesús Nogueira and Ignasi Colominas for their support in this long path called Ph.D. It has been my privilege to work closely with them. Their encouragement and support have provided me with the opportunity to explore and deeply appreciate the amazing world of computational fluid dynamics. I am sure that their inspiration, patience, knowledge, unforgettable attitude and dedicated time have been essential for the completion of this thesis and will prove valuable sources of inspiration throughout my life.

I must express my gratitude to the *Grupo de Métodos Numéricos en Ingeniería* (GMNI), specially to Manuel Casteleiro and Fermín Navarrina for their help, advice and financial support, and to José París for helping me with my "numerical issues". I feel really lucky to belong to this group, where I have found an inspiring atmosphere and the resources necessary to complete this research. My gratitude also goes to the presidents of the *Fundación de la Ingeniería Civil de Galicia*, Antonio Fernández and Juan Sanmartín, for the support provided throughout this research.

I am indebted to all my friends who have supported me over the last few years: Pablo, Javi, Cris, Sierra, and the Hankinson family. A special thanks I have enjoyed many useful and entertaining discussions with my friends and co-workers Javier Paz, Pablo Ouro, Iris González, Guillermo Lorenzo, Guillermo Vilanova and Iván Couceiro.

I also had the chance to spend six months in the *Laboratoire de dynamique des fluides* at the *Arts et Métiers ParisTech*, working with Dr. Sofiane Khelladi and his doctorands, and Dr. Jean-Camille Chassaing from the *Université Pierre et Marie Curie*. Many thanks to them for their invitation and help through these years.

My deepest gratitude to my parents Luis and Teresa, and brothers Jorge and Álvaro for their care and support. Their support and encouragements helped me to overcome my setbacks and to pursue my interests. Finally I owe much to Ana, without whose love and understanding I would not have completed this work.

If you feel that you have been left out I must have forgot to include your name, forgive me and feel gratefully acknowledged.

This work has been partially supported by the *Ministerio de Ciencia e Innovación* (grant #DPI2010-16496) and the *Ministerio de Economía y Competitividad* (grant #DPI2012-33622) of the Spanish Government and by the, *Consellería de Cultura, Educación e Ordenación Universitaria* of the *Xunta de Galicia* (grants #CN2011/002 #GRC2014-/039) cofinanced with FEDER funds, *Fundación de la Ingeniería Civil de Galicia* and the *Universidade da Coruña*.

I want to acknowledge the *Universidade da Coruña* and Inditex for providing financial support to my stay in the Arts et Métiers ParisTech in 2014.

Abstract

High-order finite volume methods based on Moving Least Squares for Computational Fluid Dynamics. Application to all-speed and incompressible flows on unstructured grids

by

Luis Ramírez Palacios

Civil Engineer

University of A Coruña

The development of high-order methods for unstructured grids remains a very active research field in Computational Fluid Dynamics (CFD). In the engineering field, most of the problems are associated with complex geometries. In these problems the use of structured meshes can lead to distorted elements that could affect the accuracy of the method. The great geometrical flexibility offered by unstructured grids makes them highly effective for dealing with complex geometries. In this context, the development of very accurate numerical methods to work on unstructured grids is very desirable.

During the past two decades, the interest in high-order methods has grown especially in certain applications where the complex flow structure and small length scales need to be adequately resolved. For example, in the simulation of turbulent flows or the propagation of acoustic waves, high-order methods are more suitable than low-order methods. However, with high-order methods there is a need to obtain high-order reconstructions of the variables.

Despite the progress made in high-order methods for CFD, common industrial simulations on unstructured meshes are usually based on second-order discretizations. These methods have been typically considered as the right choice due to their simplicity, robustness, and their effectiveness in providing a reasonably accurate solution with comparably low computational cost. However, classical second-order algorithms can be insufficient to accurately predict the flow in complicated geometries and complex physics.

This thesis presents the development of high-order numerical methods for the numerical simulation of all-speed and incompressible flows on unstructured grids. One possible application of the formulation developed in this thesis is the simulation of turbomachinery flow. The operational flow regimes associated to turbomachinery range from very low Mach numbers, which leads to nearly incompressible flows, to supersonic Mach numbers. In order to simulate the flow with a wide range of Mach numbers, a high-order finite volume density-based formulation for all-speed flows is proposed in this thesis. The high-order

reconstruction of the variables is obtained by means of a Taylor series expansion. The gradients and high-order derivatives are obtained with the Moving Least Squares approximants. It is known that density-based solvers present the so-called accuracy problem at low Mach regimes, due to excessive wrong numerical diffusion. In this thesis, it is shown that the accuracy problem is alleviated when the order of the method is increased. However, a grid dependency still remains. In order to to circumvent this dependency, several fixes have been proposed in the literature. To the author knowledge, all these fixes have been applied, at most to second-order methods. In this thesis, the use of low-Mach fixes has been extended to high-order numerical methods.

However, when all the flow in the domain presents a low Mach regime, the resolution of the compressible Navier-Stokes with an all-speed scheme may not be practical. This is motivated by the small time step required due to the large disparity between the acoustic and the flow speed. This regime is common in hydrodynamics and low speed aerodynamics, such as the flow around a tidal turbine. For these cases, the incompressibility assumption can be adopted. In this thesis, a novel high-order pressure-based formulation is proposed for the numerical resolution of the incompressible Navier-Stokes. The Semi Implicit Method for Pressure Linked Equations (SIMPLE) algorithm is used to impose the incompressibility condition on a collocated grid arrangement. The formulation is based on the use of MLS to obtain the high-order approximations of the variables. In order to avoid the checkerboard oscillations and preserve the accuracy of the scheme, MLS is employed to obtain a new Momentum Interpolation Method (MIM).

The proposed methods have been analyzed on several steady and unsteady numerical test cases with structured and unstructured mesh discretizations. The formal order of convergence is recovered and very accurate results have been obtained.

In addition, new high-order sliding mesh methods are proposed in order to simulate accurately the flow around rotating geometries. In a high-order framework, to preserve the accuracy of the numerical scheme, the simulation of rotating geometries needs to be of, at least, the same order than the numerical scheme. This is a crucial point in the development of high-order methods for the simulations of turbomachines. In this thesis, the MLS approximants are used to develop a new sliding mesh methodology, which preserves the accuracy of the numerical scheme. The accuracy and robustness of the new methodology has been investigated for several structured and unstructured mesh discretizations. The numerical results have shown that the novel numerical methodologies preserve the formal order of accuracy.

Resumen

Métodos de volúmenes finitos de alto orden basados en Mínimos Cuadrados Móviles para la Mecánica de Fluidos Computacional. Aplicación a flujos compresibles con un amplio rango de números de Mach y a flujos incompresibles en mallados no estructurados

por

Luis Ramírez Palacios
Ingeniero de Caminos, Canales y Puertos
Universidade da Coruña

La Mecánica de Fluidos Computacional (CFD del inglés Computational Fluid Dynamics) es una disciplina de la mecánica basada principalmente en la resolución de las ecuaciones de Navier-Stokes. Este sistema de ecuaciones define de manera matemática el comportamiento dinámico de un fluido. Aunque este sistema de ecuaciones se propuso en el siglo XIX, aún hoy en día, sólo unos pocos problemas pueden ser resueltos de manera analítica. Por lo tanto, la mayoría de las soluciones de las ecuaciones de Navier-Stokes deben ser obtenidas de manera numérica.

La Mecánica de Fluidos Computacional empezó a cobrar importancia a partir de los años 60, impulsada principalmente por la industria aeroespacial y el aumento de la potencia de cálculo de los computadores. Desde entonces ha evolucionado hasta convertirse en una herramienta de vital importancia para industrias donde el conocimiento de la dinámica de los flujos es necesaria para el desarrollo de nuevos dispositivos y prototipos. Por ejemplo, el diseño de vehículos más eficientes o menos ruidosos en el sector aeronáutico y automovilístico.

A día de hoy, la mayoría de los métodos numéricos empleados habitualmente en la industria se basan generalmente en discretizaciones espaciales de segundo orden. Estos métodos se han considerado como la mejor opción debido a su simplicidad, robustez, y su eficacia para obtener una solución razonablemente precisa con un bajo coste computacional. Sin embargo, los métodos de segundo orden pueden no ser suficientes para predecir el flujo de una manera precisa en geometrías complicadas o en problemas en los que la física sea compleja, como por ejemplo, problemas de acústica o problemas con flujos turbulentos. Debido a esto, el interés por los métodos numéricos de alta precisión ha aumentado durante las últimas dos décadas. Además, los métodos de alto orden ofrecen el potencial de reducir significativamente el coste computacional con respecto a los métodos de bajo orden, ya que es posible obtener resultados con la misma precisión empleando una resolución de malla

menor.

El desarrollo de métodos de alto orden para mallas no estructuradas es un área de investigación muy activo en la mecánica de fluidos computacional. En la mayoría de los problemas habituales en ingeniería, las geometrías asociadas suelen ser complejas, y la construcción de una malla estructurada en estas geometrías puede dar lugar a elementos distorsionados que podrían afectar la precisión del método. La gran flexibilidad geométrica ofrecida por los mallados no estructurados hace de este tipo de mallas una opción eficaz para discretizar geometrías complejas. En este contexto es necesario el desarrollo de métodos numéricos de gran precisión para mallados no estructurados.

En esta tesis se desarrollan métodos numéricos de alto orden para la simulación numérica en mallados no estructurados de flujos compresibles con un amplio rango de números de Mach y para la simulación numérica de flujos incompresibles. Una posible aplicación de las formulaciones desarrolladas en esta tesis es la simulación del flujo en una turbomáquina.

Las turbomáquinas están presentes bajo muchas formas en nuestra vida cotidiana, desde las bombas centrífugas y los ventiladores de uso común hasta las grandes turbinas hidráulicas de las centrales hidroeléctricas, siendo un elemento fundamental en la generación de energía y en numerosos procesos industriales. La simulación numérica de las turbomáquinas es de gran interés para comprender el complejo comportamiento del flujo en el interior de estas y así poder modificar el diseño para mejorar su eficiencia. Los métodos numéricos no sólo permiten la simulación del flujo en una turbina, sino que si son lo suficientemente precisos, permiten analizar, por ejemplo, la propagación del ruido aerodinámico generado.

Los regímenes de operación de flujo asociados a las turbomáquinas son muy amplios, comprendiendo desde números de Mach muy bajos, lo que permite que los flujos puedan considerarse incompresibles (como el flujo en una turbina hidráulica o en una bomba centrífuga), hasta altos números de Mach, con regímenes supersónicos en el fluido, como por ejemplo, el flujo en una turbina de gas o de vapor.

Con el objetivo de simular los flujos para un amplio rango de números de Mach, en esta tesis se propone una formulación de volúmenes finitos de alto orden para la resolución de las ecuaciones de Navier-Stokes compresibles mediante algoritmos basados en la densidad. La formulación propuesta está basada en el método generalizado de Godunov, donde las reconstrucciones de alto orden de las variables se obtienen mediante desarrollos en serie de Taylor. En la práctica, la obtención de métodos de volúmenes finitos de alto orden en mallados no estructurados está limitada al cálculo de los gradientes y las sucesivas derivadas necesarias para la reconstrucción de alto orden de las variables. En esta tesis, estos valores

se obtienen de una manera muy precisa con el método de Mínimos Cuadrados Móviles (MLS de su acrónimo en inglés), que permite obtener las derivadas necesarias a partir de una serie de puntos dispersos, dando como resultado el método FV-MLS.

Es conocido que, en un contexto de volúmenes finitos, la resolución de las ecuaciones de Euler o Navier-Stokes mediante algoritmos basados en la densidad para flujos con números de Mach bajos presenta el llamado problema de precisión ("*accuracy problem*"), que puede dar lugar a la obtención de soluciones de flujo no físicas. Este problema se debe a una excesiva difusión introducida por los flujos numéricos.

En esta tesis se demuestra que el problema de precisión se alivia cuando se aumenta el orden del método numérico. Sin embargo, el método sigue presentando una dependencia con el mallado necesario para obtener una solución física para un determinado número de Mach. Con el fin de eliminar esta dependencia, diversos autores han propuesto varias modificaciones a los flujos numéricos convencionales. Sin embargo, a conocimiento del autor, todas estas correcciones no se han aplicado nunca a métodos de alto orden. En esta tesis, el uso de las modificaciones de los flujos numéricos de Roe y Rusanov para bajos números de Mach en métodos de primer orden se ha extendido a métodos numéricos de alto orden (> 2), consiguiendo soluciones físicas para números de Mach muy bajos sin dependencia de la malla.

Por otra parte, una formulación de volúmenes finitos de alto orden para la simulación numérica de flujos compresibles necesita incluir un mecanismo de limitación para los regímenes de flujos transónicos y supersónicos, donde la aparición de ondas de choque es un fenómeno frecuente. Uno de los métodos de estabilización más habituales en volúmenes finitos de alto orden son los limitadores de pendiente. Idealmente, el limitador no debería activarse en regiones donde no hay ondas de choque para que la precisión del método no disminuya. Sin embargo, en esta tesis se ha comprobado que el uso de limitadores no diferenciables junto con las modificaciones para bajo Mach (propuestas para métodos de orden bajo) destruye la precisión de estos métodos. Además, tanto los limitadores diferenciables (limitador de Venkatakrishnan), como los no diferenciables (limitadores de Barth and Jespersen y Van Albada), se activan de manera innecesaria a bajos número de Mach, provocando también una pérdida de precisión en el método numérico. Para solucionar estos problemas se propone el uso de un sensor basado en MLS para detectar ondas de choque.

Cuando todo el flujo en el dominio presenta un régimen de bajo Mach, la resolución de las ecuaciones de Navier-Stokes en su versión compresible puede no ser de utilidad práctica. Esto está motivado por el pequeño paso de tiempo requerido por la gran disparidad entre la velocidad de las ondas acústicas y la velocidad del flujo. Este régimen es común en problemas de hidrodinámica y aerodinámica a baja velocidad, tales como el flujo alrededor

de una turbina eólica o en las turbinas utilizadas para la extracción de energía mareomotriz. Para estos casos, es posible adoptar la hipótesis de incompresibilidad del fluido.

En esta tesis se desarrolla una formulación de alto orden para resolver de manera numérica las ecuaciones de Navier-Stokes en su versión incompresible. Al asumir la hipótesis de incompresibilidad, las ecuaciones de continuidad y de momentos se desacoplan. Este fenómeno se debe a la ausencia de un término transitorio en la ecuación de continuidad, por lo que ésta se puede considerar como una restricción que el campo de velocidades debe cumplir. El principal problema para obtener soluciones numéricas de las ecuaciones de Navier-Stokes incompresibles reside en acoplar los cambios en el campo de velocidades con los cambios en el campo de presiones. Debido al desacoplamiento de las ecuaciones de continuidad y de momentos. Para resolver este problema hay tres aproximaciones principales:

- Los métodos basados en presión.
- Los métodos del tipo "compresibilidad artificial".
- Los métodos basados en variables derivadas, como los métodos basados en la vorticidad.

En un contexto de volúmenes finitos para flujos incompresibles, hay dos tipos de disposición de variables en las mallas:

- La discretización colocada.
- La discretización escalonada ("*staggered*").

Los mallados basados en una disposición colocada de las variables las ubican en un mismo punto de la celda (habitualmente en el centroide), siendo habitualmente empleados en mallados no estructurados. Sin embargo, los mallados colocados pueden presentar oscilaciones del tipo "*checkerboard*" que afecten a la precisión del método debido a la interpolación centrada de la presión. Con el fin de evitar las oscilaciones, Rhie y Chow propusieron el método de interpolación de momentos (MIM), una interpolación lineal que evita el desacoplamiento de la velocidad con la presión. En esta tesis, con el fin de evitar las oscilaciones y preservar la precisión del esquema, el método MLS se emplea para obtener un nuevo método de alto orden de interpolación de momentos. El algoritmo "*Semi Implicit Method for Pressure Linked Equations*" (SIMPLE), basado en la presión, se utiliza para imponer la condición de incompresibilidad en una discretización de mallados colocados.

Los mallados tipo "*staggered*" sitúan las variables en ubicaciones diferentes, las velocidades se sitúan en las aristas de las celdas y la presión y la densidad en el centroide. Esta

disposición evita el uso de interpolaciones y por lo tanto no presentan "*checkerboard*". Sin embargo, esta disposición de las variables presenta dificultades en su aplicación a mallados no estructurados.

Con el fin de simular de manera precisa el flujo alrededor de geometrías móviles, como pueden ser los álabes en una turbina, en esta tesis se han desarrollado técnicas de alto orden para métodos con mallas móviles. Las técnicas desarrolladas están basadas en los algoritmos "*sliding mesh*" que consisten en discretizar el dominio de cálculo con diferentes regiones de malla en movimiento relativo entre ellas. Por ejemplo, en una turbomáquina, se pueden diferenciar dos zonas: el estátor (que permanece fijo) y el rotor (que presenta un movimiento rotativo). Dado que existe un movimiento relativo entre las dos regiones, es necesario transferir la información de una región a otra. En un contexto de alto orden, y con el fin de acoplar las diferentes regiones y mantener la precisión del esquema numérico, la transferencia de información entre diferentes regiones debe ser, al menos, del mismo orden que el esquema numérico. Este punto es crucial para desarrollar métodos de alto orden para la simulación de turbomáquinas. El método de MLS se emplea para desarrollar nuevas formulaciones de "*sliding mesh*" que conserven la precisión del esquema numérico.

En esta tesis, se han considerado dos enfoques distintos: el primer enfoque está basado en los métodos más habituales de "*sliding mesh*", obtenidos a partir de consideraciones geométricas; mientras que el segundo método aprovecha las propiedades de aproximación del método MLS para desarrollar un método más flexible y más fácil de implementar en cualquier código de volúmenes finitos existente, ya que evita el cálculo de intersecciones. La precisión y robustez de las nuevas metodologías propuestas ha sido investigado en mallados estructurados y no estructurados.

De los trabajos de investigación realizados en esta tesis se han obtenido las siguientes conclusiones:

1. Se ha desarrollado un método de volúmenes finitos de alto orden para la resolución de las ecuaciones de Navier-Stokes compresibles en geometrías complejas para un amplio rango de números de Mach. Los principales ingredientes de esta formulación de volúmenes finitos son el método MLS, una modificación en los flujos numéricos para solucionar el problema de precisión cuando el número de Mach es pequeño y una estrategia de limitación de pendiente junto con un sensor de choques basado en el método MLS. Se han obtenido las siguientes conclusiones particulares:

   (a) Se ha demostrado que aumentando el orden de la discretización espacial se alivia el problema de precisión a bajos números de Mach. Sin embargo, el aumento del orden del esquema no resuelve completamente este problema.

(b) Para resolver completamente el problema de precisión a bajo Mach, se ha propuesto el uso de esquemas de alto orden, junto con las correcciones para bajo Mach desarrolladas para métodos de orden uno. Con esto, se consigue conservar la precisión de la solución a números de Mach tan bajos como 10^{-6}, y mantienen el orden de convergencia.

(c) Se ha demostrado que el limitador de pendiente puede permanecer activo para números de Mach pequeños, en regiones donde no hay ondas de choque. La activación de un limitador de pendiente no diferenciable en regiones con un flujo a bajo Mach destruye la precisión aportada por las correcciones de los flujos numéricos para bajo Mach.

(d) A fin de mantener la precisión del esquema de orden superior, se propone la aplicación de un sensor de ondas de choque basado en aproximaciones MLS. Este sensor impide la activación innecesaria del limitador de pendiente, evitando así la pérdida de precisión descrita en el apartado anterior en regiones con bajo Mach.

(e) La precisión del método propuesto se ha probado mediante varios ejemplos numéricos estacionarios y no estacionarios. De los resultados numéricos obtenidos se demuestra que el nuevo método recupera la solución física y que obtiene el orden de convergencia esperado.

2. Se ha desarrollado un esquema de alto orden de volúmenes finitos para la resolución de las ecuaciones Navier-Stokes en su versión incompresible para geometrías complejas. La condición de incompresibilidad se impone de manera iterativa en cada paso de tiempo utilizando el algoritmo SIMPLE. El método de mínimos cuadrados móviles se emplea con el fin de obtener las aproximaciones y derivadas necesarias para obtener un método numérico de alto orden en mallados colocados.

   (a) La discretización lineal habitualmente empleada en el método de interpolación de momentos (MIM) se ha extendido a alto orden mediante el método de mínimos cuadrados móviles con el fin de conservar el orden de convergencia del esquema.

   (b) El método propuesto ha sido validado en varios test numéricos estacionarios y no estacionarios. De los resultados numéricos se demuestra que el nuevo método obtiene el orden de convergencia esperado y los resultados con el nuevo esquema mejoran a los obtenidos mediante los métodos más habituales en la industria.

3. Se han desarrollado tres métodos de "*sliding mesh*" de alto orden basados en el método

de mínimos cuadrados móviles. Se han obtenido las siguientes conclusiones particulares:

(a) La precisión y robustez de la nueva metodología se ha investigado en diversos casos test numéricos, tanto en flujos incompresibles como en compresibles. Los resultados numéricos han demostrado que los métodos de "*sliding mesh*" propuestos son capaces de conservar el orden esperado de convergencia del esquema.

(b) Se ha desarrollado una metodología "*sliding mesh*" que no precisa el cálculo de intersecciones. Esta nueva técnica mantiene el orden del esquema y obtiene resultados tan precisos como las técnicas basadas en el cálculo de intersecciones entre celdas.

(c) Una clara ventaja de las técnicas de "*sliding mesh*" propuestas es que el mismo esquema de discretización es empleado para la discretización de todo el dominio de cálculo, evitando la aparición de soluciones espúreas entre las diferentes regiones.

xxx

Resumo

Métodos de volumes finitos de alta orde baseados en Mínimos Cadrados Móbiles para a Mecánica de Fluídos Computacional. Aplicación a fluxos compresibles cun amplo rango de números de Mach e a fluxos incompresibles en mallados non estruturados

por

Luis Ramírez Palacios
Enxeñeiro de Camiños, Canais e Portos
Universidade da Coruña

A Mecánica de Fluídos Computacional (CFD do inglés Computational Fluid Dynamics) é unha disciplina da mecánica baseada principalmente na resolución das ecuacións de Navier-Stokes. Este sistema de ecuacións define de xeito matemático o comportamento dinámico dun fluído. Aínda que este sistema de ecuacións foi proposto no século XIX, aínda hoxe en día, só uns poucos problemas poden ser resoltos de xeito analítico. Polo tanto, a maioría das solucións das ecuacións de Navier-Stokes deben ser obtidas de xeito numérico.

A Mecánica de Fluídos Computacional comezou a cobrar importancia a partires dos anos 60, impulsada principalmente pola industria aeroespacial e o aumento da potencia de cálculo dos computadores. Dende entón evolucionou ata converterse nunha ferramenta de vital importancia para industrias onde o coñecemento da dinámica dos fluxos é necesaria para o desenvolvemento de novos dispositivos e prototipos. Por exemplo, no deseño de vehículos máis eficientes ou menos ruidosos no sector aeronáutico e automobilístico.

A día de hoxe, a maioría dos métodos numéricos empregados habitualmente na industria baséanse xeralmente en discretizacións espaciais de segunda orde. Estes métodos foron considerados como a mellor opción debido á súa simplicidade, robustez, e a súa eficacia para obter unha solución razoablemente precisa cun baixo custo computacional. Non obstante, os métodos de segunda orde poden non ser suficientes para predicir o fluxo dun xeito preciso en xeometrías complicadas ou en problemas nos que a física sexa complexa, como por exemplo, problemas de acústica ou problemas con fluxos turbulentos. Debido a isto, o interese polos métodos numéricos de alta precisión aumentou durante as últimas dúas décadas. Ademais, os métodos de alta orde ofrecen o potencial de reducir significativamente o custo computacional con respecto aos métodos de baixa orde, xa que é posible obter resultados coa mesma precisión empregando unha resolución de malla menor.

O desenvolvemento de métodos de alta orde para mallas non estruturadas é unha área de investigación moi activa na mecánica de fluídos computacional. Na maioría dos

problemas habituais en enxeñaría, as xeometrías asociadas adoitan ser complexas, e a construción dunha malla estruturada nestas xeometrías pode dar lugar a elementos distorsionados que poderían afectar a precisión do método. A gran flexibilidade xeométrica ofrecida polos mallados non estruturados fai deste tipo de mallas unha opción eficaz para discretizar xeometrías complexas. Neste contexto é necesario o desenvolvemento de métodos numéricos de gran precisión para mallados non estruturados.

Nesta tese desenvólvense métodos numéricos de alta orde para a simulación numérica en mallados non estruturados de fluxos compresibles cun amplo rango de números de Mach e para a simulación numérica de fluxos incompresibles. Unha posible aplicación das formulacións desenvolvidas nesta tese é a simulación do fluxo nunha turbomáquina.

As turbomáquinas están presentes baixo moitas formas na nosa vida cotiá, dende as bombas centrífugas e os ventiladores de uso común ata as grandes turbinas hidráulicas das centrais hidroeléctricas, sendo un elemento fundamental na xeración de enerxía e en numerosos procesos industriais. A simulación numérica das turbomáquinas é de grande interese para comprender o complexo comportamento do fluxo no interior destas e así poder modificar o deseño para mellorar a súa eficiencia. Os métodos numéricos non só permiten a simulación do fluxo nunha turbina, senón que se son o suficientemente precisos, permiten analizar, por exemplo, a propagación do ruído aerodinámico xerado.

As condicións de operación de fluxo asociados ás turbomáquinas son moi amplos, comprendendo dende números de Mach moi baixos, o que permite que os fluxos poidan considerarse incompresibles (como o fluxo nunha turbina hidráulica ou nunha bomba centrífuga), ata altos números de Mach, con réximes supersónicos no fluído, como por exemplo, o fluxo unha turbina de gas ou de vapor.

Co obxectivo de simular os fluxos para un amplo rango de números de Mach, nesta tese proponse unha formulación de volumes finitos de alta orde para a resolución das ecuacións de Navier-Stokes compresibles mediante algoritmos baseados na densidade. A formulación proposta está baseada no método xeneralizado de Godunov, onde as reconstrucións de alta orde das variables se obteñen mediante desenvolvementos en serie de Taylor. Na práctica, a obtención de métodos de volumes finitos de alta orde en mallados non estruturados está limitada ao cálculo dos gradientes e as sucesivas derivadas necesarias para a reconstrución de alta orde das variables. Nesta tese, estes valores obtéñense dun xeito moi preciso co método de Mínimos Cadrados Móbiles (MLS do seu acrónimo en inglés), que permite obter as derivadas necesarias a partir dunha serie de puntos dispersos, dando como resultado o método FV-MLS.

É coñecido que, nun contexto de volumes finitos, a resolución das ecuacións de Euler ou Navier-Stokes mediante algoritmos baseados na densidade para fluxos con números

de Mach baixos presenta o chamado problema de precisión ("*accuracy problem*"), que pode dar lugar á obtención de solucións de fluxo non físicas. Este problema é debido a unha excesiva difusión introducida polos fluxos numéricos.

Nesta tese demóstrase que o problema de precisión se alivia ao aumentar a orde do método numérico. Non obstante, o método segue presentando unha dependencia co mallado necesario para obter unha solución física para un determinado número de Mach. Co fin de eliminar esta dependencia, diversos autores propuxeron varias modificacións aos fluxos numéricos convencionais. Non obstante, a coñecemento do autor, todas estas correccións non foron aplicadas nunca a métodos de alta orde. Nesta tese, o uso das modificacións dos fluxos numéricos de Roe e Rusanov para baixos números de Mach en métodos de primeira orde estendeuse a métodos numéricos de alta orde (> 2), conseguindo solucións físicas para números de Mach moi baixos sen dependencia da malla.

Por outra parte, unha formulación de volumes finitos de alta orde para a simulación numérica de fluxos compresibles necesita incluír un mecanismo de limitación para os réximes de fluxos transónicos e supersónicos, onde a aparición de ondas de choque é un fenómeno frecuente. Un dos métodos de estabilización máis habituais en volumes finitos de alta orde son os limitadores de pendente. Idealmente, o limitador non debería activarse en rexións onde non hai ondas de choque para que a precisión do método non diminúa. Non obstante, nesta tese comprobouse que o uso de limitadores non diferenciables xunto coas modificacións para baixo Mach (propostas para métodos de orde baixa) destrúe a precisión destes métodos. Ademais, tanto os limitadores diferenciables (limitador de Venkatakrishnan), coma os non diferenciables (limitadores de Barth and Jespersen e Van Albada), actívanse de xeito innecesario a baixos número de Mach, provocando tamén unha perda de precisión no método numérico. Para solucionar estes problemas proponse o uso dun sensor baseado en MLS para detectar ondas de choque.

Cando todo o fluxo no dominio presenta un réxime de baixo Mach, a resolución das ecuacións de Navier-Stokes na súa versión compresible pode non ser de utilidade práctica. Isto está motivado polo pequeno paso de tempo requirido pola gran disparidade entre a velocidade das ondas acústicas e a velocidade do fluxo. Estas condicións son comúns en problemas de hidrodinámica e aerodinámica a baixa velocidade, tales como o fluxo ao redor dunha turbina eólica ou nas turbinas utilizadas para a extracción de enerxía mareomotora. Para estes casos, é posible adoptar a hipótese de incompresibilidade do fluído.

Nesta tese desenvólvese unha formulación de alta orde para resolver de xeito numérico as ecuacións de Navier-Stokes na súa versión incompresible. Ao asumir a hipótese de incompresibilidade, as ecuacións de continuidade e de momentos desadáptanse. Este fenómeno é debido á ausencia dun termo transitorio na ecuación de continuidade, polo que

esta pódese considerar como unha restrición que o campo de velocidades debe cumprir. O principal problema para obter solucións numéricas das ecuacións de Navier-Stokes incompresibles reside en adaptar os cambios no campo de velocidades cos cambios no campo de presións. Debido ao desacoplamento das ecaucións de continuidades e de momentos. Para resolver este problema hai tres aproximacións principais:

- Os métodos baseados na presión.
- Os métodos do tipo "compresibilidade artificial".
- Os métodos baseados en variables derivadas, como os métodos baseados na vorticidade.

Nun contexto de volumes finitos para fluxos incompresibles, hai dous tipos de disposición de variables nas mallas:

- A discretización colocada.
- A discretización graduada ("*staggered*").

Os mallados baseados nunha disposición colocada sitúan ás variables nun mesmo punto da cela (habitualmente na centroide), sendo empregados habitualmente en mallados non estruturados. Non obstante, os mallados colocados poden presentar oscilacións do tipo "*checkerboard* " que afecten á precisión do método debido á interpolación centrada da presión. Co fin de evitar as oscilacións, Rhie e Chow propuxeron o método de interpolación de momentos (MIM), unha interpolación lineal que evita o desacoplamento da velocidade coa presión. Nesta tese, co fin de evitar as oscilacións e preservar a precisión do esquema, o método MLS emprégase para obter un novo método de alta orde de interpolación de momentos. O algoritmo "*Semi Implicit Method for Pressure Linked Equations*" (SIMPLE), baseado na presión, utilízase para impoñer a condición de incompresibilidade nunha discretización de mallados colocados.

Os mallados tipo "*staggered*" sitúan as variables en situacións diferentes, as velocidades sitúanse nas arestas das celas e a presión e a densidade na centroide. Esta disposición evita o uso de interpolacións e polo tanto non presentan "*checkerboard*". Non obstante, esta disposición das variables presenta dificultades na súa aplicación a mallados non estruturados.

Co fin de simular de xeito preciso o fluxo arredor de xeometrías móbiles, como poden ser as penlas nunha tubina, nesta tese desenvolvéronse técnicas de alta orde para métodos con mallas móbiles. As técnicas desenvolvidas están baseadas nos algoritmos "*sliding mesh*" que consisten en discretizar o dominio de cálculo con diferentes rexións de malla

en movemento relativo entre elas. Por exemplo, nunha turbomáquina, pódense diferenciar dúas zonas: o estátor (que permanece fixo) e o rotor (que presenta un movemento rotativo). Dado que existe un movemento relativo entre as dúas rexións, é necesario transferir a información dunha rexión a outra. Nun contexto de alta orde, e co fin de adaptar as diferentes rexións e manter a precisión do esquema numérico, a transferencia de información entre diferentes rexións debe ser, polo menos, da mesma orde que o esquema numérico. Este punto é crucial para desenvolver métodos de alta orde para a simulación de turbomáquinas. O método de MLS emprégase para desenvolver novas formulacións de "*sliding mesh*" que conserven a precisión do esquema numérico.

Nesta tese, consideráronse dous enfoques distintos: o primeiro enfoque está baseado nos métodos máis habituais de "*sliding mesh*" obtidas a partires de consideracións xeométricas; mentres que o segundo método aproveita as propiedades de aproximación do método MLS para desenvolver un método máis flexible e máis doado de implementar en calquera código de volumes finitos existente, xa que evita o cálculo de interseccióǹs. A precisión e robustez das novas metodoloxías propostas foi investigado en mallados estructurados e non estruturados.

Dos traballos de investigación realizados nesta tese obtivéronse as seguintes conclusións:

1. Desenvolveuse un método de volumes finitos de alta orde para a resolución das ecuacións de Navier-Stokes compresibles en xeometrías complexas para un amplo rango de números de Mach. Os principais ingredientes desta formulación de volumes finitos son o método MLS, unha modificación nos fluxos numéricos para solucionar o problema de precisión cando o número de Mach é pequeno e unha estratexia de limitación de pendente xunto cun sensor de choques baseado no método MLS. Obtivéronse as seguintes conclusións particulares:

   (a) Demostrouse que aumentando a orde da discretización espacial se alivia o problema de precisión a baixos números de Mach. Non obstante, o aumento da orde do esquema non resolve completamente este problema.

   (b) Para resolver completamente o problema de precisión a baixo Mach, propúxose o uso de esquemas de alta orde, xunto coas correccións para baixo Mach desenvolvidas para métodos de orde un. Con isto, conséguese conservar a precisión da solución a números de Mach tan pequenos como 10^{-6}, mantendo a orde de converxencia.

   (c) Demostrouse que o limitador de pendente pode permanecer activo para números de Mach pequenos, en rexións onde non hai ondas de choque. A activación

dun limitador de pendente non diferenciable en rexións cun fluxo a baixo Mach destrúe a precisión obtida polas correccións dos fluxos numéricos para baixo Mach.

(d) Co fin de manter a precisión do esquema de orde superior, proponse a aplicación dun sensor de ondas de choque baseado en aproximacións MLS. Este sensor impide a activación innecesaria do limitador de pendente, evitando así a perda de precisión descrita no apartado anterior en rexións con baixo Mach.

(e) A precisión do método proposto probouse mediante varios exemplos numéricos estacionarios e non estacionarios. Dos resultados numéricos obtidos, demóstrase que o novo método recupera a solución física e que obtén a orde de converxencia esperada.

2. Desenvolveuse un esquema de alta orde de volumes finitos para a resolución das ecuacións Navier-Stokes na súa versión incompresible para xeometrías complexas. A condición de incompresibilidade imponse de xeito iterativo en cada paso de tempo utilizando o algoritmo SIMPLE. O método de mínimos cadrados móbiles emprégase co fin de obter as aproximacións e derivadas necesarias para obter un método numérico de alta orde en mallados colocados.

   (a) A discretización lineal habitualmente empregada no método de interpolación de momentos (MIM) estendeuse a alta orde mediante o método de mínimos cadrados móbiles co fin de conservar a orde de converxencia do esquema.

   (b) O método proposto foi validado en varios test numéricos estacionarios e non estacionarios. Dos resultados numéricos demóstrase que o novo método obtén a orde de converxencia esperada e os resultados co novo esquema melloran aos obtidos mediante os métodos máis habituais na industria.

3. Desenvolvéronse tres métodos de "*sliding mesh*" de alta orde baseados no método de mínimos cadrados móbiles. Obtivéronse as seguintes conclusións particulares:

   (a) A precisión e robustez da nova metodoloxía investigouse en diversos casos test numéricos, tanto en fluxos incompresibles coma en compresibles. Os resultados numéricos demostraron que os métodos de "*sliding mesh*" propostos son capaces de conservar a orde esperada de converxencia do esquema.

   (b) Desenvolveuse unha metodoloxía "*sliding mesh*" que non precisa o cálculo de interseccióńs. Esta nova técnica mantén a orde do esquema e obtén resutlados tan precisos como as técnicas baseadas no cálculo de interseccióńs entre celdas.

(c) Unha clara vantaxe das técnicas de "*sliding mesh*" propostas é que o mesmo esquema de discretización é empregado para a discretización de todo o dominio de cálculo, evitando a aparición de solucións espurias entre as diferentes rexións.

Chapter 1

Introduction.

1.1 Introduction

From the air we breathe to the water we drink, from skies to the oceans we travel, fluid flow is a fundamental phenomenon of nature and human life. As we have sought to better understand and manipulate our environment, the desire to study the fluid phenomena is a natural one. There are two main complementary ways to study it, experimentally and theoretically. Leonhard Euler published in 1759 the first theoretical equations that describe the motion for a fluid [52]. Euler's idea to express the fluid dynamics in the form of partial differential equations was a major breakthrough, but a practical shortcoming of his model is that it does not consider friction forces. In 1822, Claude Navier [143] introduced a more advanced system of equations that solved the Euler's equations shortcoming, but it only worked for incompressible flows. Later on, George Stokes came in 1845 with the system of equations for compressible fluids [197]. The system of equations introduced by Stokes is the today called Navier-Stokes equations. The Navier-Stokes equations are generally accepted as the partial differential equations (PDE's) governing the flow of newtonian fluids in a continuum regime. They can be written in two-dimensional form as

$$\partial_t \boldsymbol{U} + \nabla \cdot \boldsymbol{\mathcal{F}}^{\mathcal{H}}(\boldsymbol{U}) - \nabla \cdot \boldsymbol{\mathcal{F}}^{\mathcal{V}}(\boldsymbol{U}, \boldsymbol{\nabla U}) = 0 \tag{1.1}$$

where $\boldsymbol{\mathcal{F}}^{\mathcal{H}}(\boldsymbol{U}) = (\boldsymbol{\mathcal{F}}^{\mathcal{H}}_x, \boldsymbol{\mathcal{F}}^{\mathcal{H}}_y)$ defines the non-viscous fluxes, $\boldsymbol{\mathcal{F}}^{\mathcal{V}}(\boldsymbol{U}, \boldsymbol{\nabla U}) = (\boldsymbol{\mathcal{F}}^{\mathcal{V}}_x, \boldsymbol{\mathcal{F}}^{\mathcal{V}}_y)$ defines the viscous fluxes and $\boldsymbol{U}(\boldsymbol{x}, t)$ describes the conservative variables, expressed as

$$\boldsymbol{U}(\boldsymbol{x}, t) = \left\{ \begin{array}{c} \rho \\ \rho v_x \\ \rho v_y \\ \rho E \end{array} \right\}. \tag{1.2}$$

The fluxes are expressed as

$$\boldsymbol{\mathcal{F}}_x^{\mathcal{H}} = \begin{pmatrix} \rho v_x \\ \rho v_x^2 + p \\ \rho v_x v_y \\ \rho v_x H \end{pmatrix} \quad \boldsymbol{\mathcal{F}}_y^{\mathcal{H}} = \begin{pmatrix} \rho v_y \\ \rho v_x v_y \\ \rho v_y^2 + p \\ \rho v_y H \end{pmatrix} \tag{1.3}$$

$$\boldsymbol{\mathcal{F}}_x^{\mathcal{V}} = \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ v_x \tau_{xx} + v_y \tau_{xy} - q_x \end{pmatrix} \quad \boldsymbol{\mathcal{F}}_y^{\mathcal{V}} = \begin{pmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ v_x \tau_{xy} + v_y \tau_{yy} - q_y \end{pmatrix} \tag{1.4}$$

where ρ is the density, $\boldsymbol{v} = (v_x, v_y)$ the velocity, μ is the viscosity of the fluid, H the enthalpy, E is the total energy, τ_{ij} is the viscous stress and $\boldsymbol{q} = (q_x, q_y)$ is a thermal flux. If the viscous flux, $\boldsymbol{\mathcal{F}}^{\mathcal{V}}$, is neglected, the Euler equations are obtained.

This set of equations describes the physics of a large number of fluid problems. However, the analytical mathematical solution of the equations is too far to be reached. Even nowadays, the Navier-Stokes can be solved analytically only for very few simple cases. Therefore, most of the flow problems need to be solved numerically. Numerical methods, like the finite volume method, the finite difference method or the finite element method, are different discretization tools developed in the XX century to solve systems of equations. The problem domain needs to be split into a discrete number of elements or subdomains where the discretized governing equations are solved. This is the basic idea behind the Computational Fluid Dynamics (CFD). The conceptual origin of CFD can be traced back to 1917, when Richardson attempted to predict weather for an eight-hour period. This attempt took him six weeks and ended in failure [175]. In Richardson's proposal, the atmosphere that covers the globe was divided into a grid, as shown in Figure 1.1, in which the primitive differential equations were solved manually. The enormous calculation requirements of his model led Richardson to propose what he called the *forecast-factory*, that he described in [175] as:

"Imagine a large hall like a theatre, except that the circles and galleries go right round through the space usually occupied by the stage. The walls of this chamber are painted to form a map of the globe. The ceiling represents the north polar regions, England is in the gallery, the tropics in the upper circle, Australia on the dress circle and the Antarctic in the pit.

A myriad computers are at work upon the weather of the part of the map where each sits, but each computer attends only to one equation or part of an equation. The work of each region is coordinated by an official of higher rank. Numerous little 'night signs' display

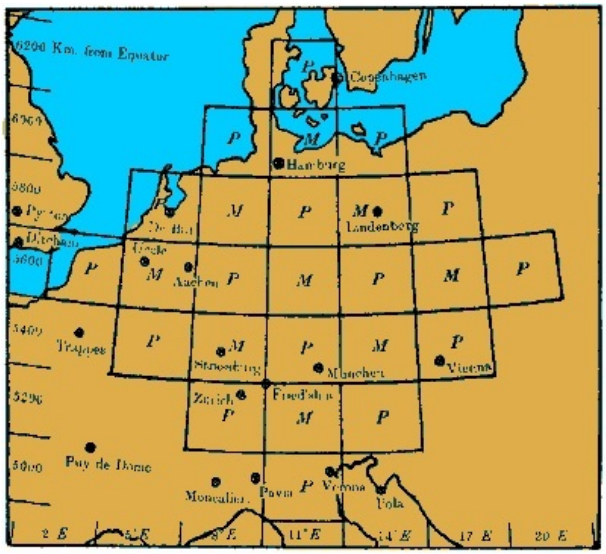


Figure 1.1: Richardson's map grid. From [175].

the instantaneous values so that neighbouring computers can read them. Each number is thus displayed in three adjacent zones so as to maintain communication to the North and South on the map."

In Figure 1.2, an illustration shows a representation of the *forecast-factory*. Note that when Richardson refers to computers, he refers to the original sense of the word, people who did computations with a mechanic calculator.

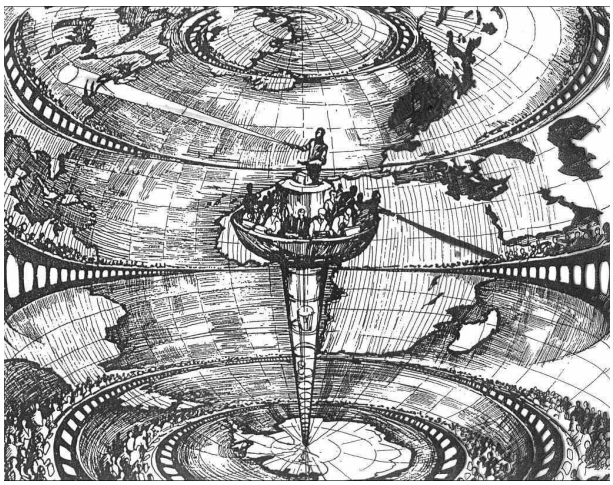


Figure 1.2: Richardson's forecast-factory. From [128].

In 1933, Thom obtained the numerical solution for the viscous flow past a cylinder,

considered the earliest numerical solution on the basis of the Navier-Stokes equations [201]. However, the real beginning of CFD era should be dated in the 1940s with the work of Kopal, who compiled massive tables of the supersonic flow over sharp cones by numerically solving the governing differential equations [106, 225]. Since the late 1960s, there has been a considerable growth in the development and application of CFD to all aspects of fluid dynamics.

The Navier-Stokes equations describe a large number of fluid problems. These problems can be classified in several ways. One possibility is to classify them according to the ratio of the flow velocity to the speed of the sound in the fluid. This ratio is called the Mach number $M = |\boldsymbol{v}|/c$, where $|\boldsymbol{v}|$ is the flow velocity and c is the speed of the sound in the fluid. This classification is depicted in Figure 1.3.

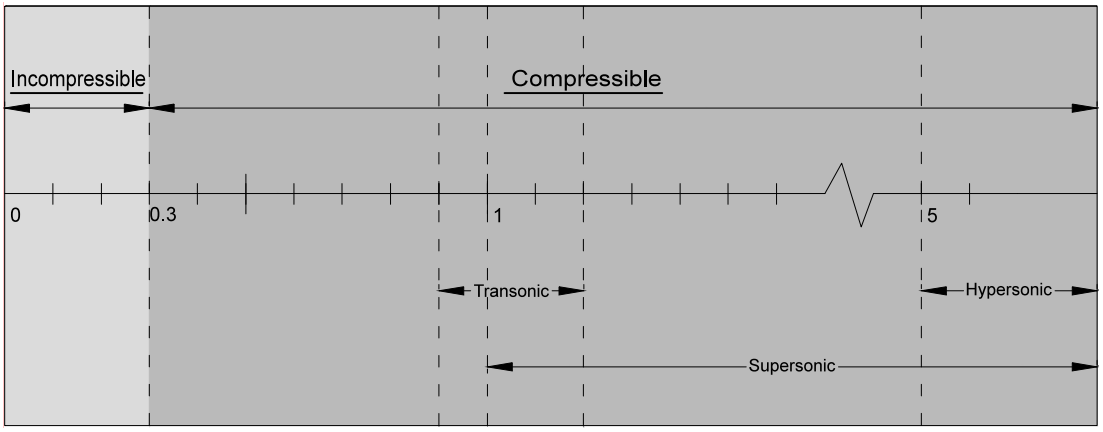


Figure 1.3: Classification of the flow as function of the Mach number. The limits are taken from [59].

The Mach number is related to the compressibility of a flow, that measures the relative change in the density field due to the variation of the pressure. Several authors [59, 182, 220] situate the compressibility limit at $M = 0.3$. At this velocity, the maximum relative change in the density field is around 5%, and therefore the compressibility effects can be neglected.

Another possibility is to classify the problems according to the Reynolds number (Re), it is a non-dimensional number defined as the ratio of the inertial forces to the viscous forces. Mathematically, this statement can be written as

$$Re = \frac{\rho v_\infty D}{\mu} \tag{1.5}$$

where ρ is the density of the fluid, v_∞ is the reference velocity and D is the characteristic

length. The Reynolds number is used to characterize the regime of a flow. For example, in Figure 1.4 it is shown the different patterns of the flow around a cylinder for different Reynolds numbers.

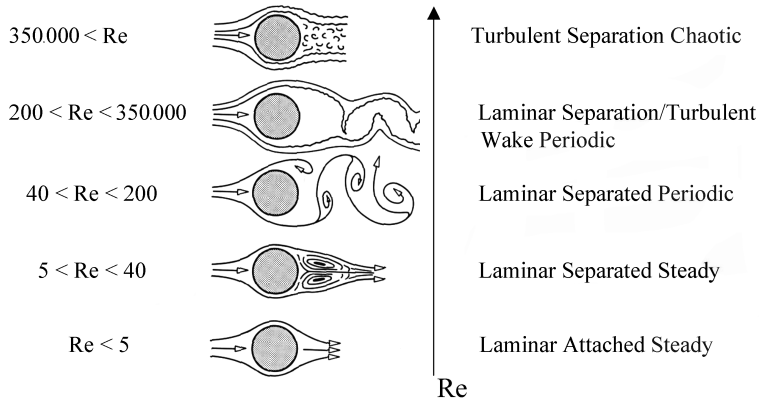


Figure 1.4: Schematic representation of the flow patterns around a cylinder for different Reynolds number. From [84].

As it was mentioned before, the analytical solution for complex problems is still far from be achieved. Thus, since the 1960s, numerical methods have become a fundamental tool in the design process that has improved our ability to build more efficient airplanes, cars, ships... The use of CFD in the industrial processes, has led to reductions in the cost of final products and development processes and has reduced the need for physical experimentation. For example, in the aircraft industry, it has become an indispensable tool on the development of new products.

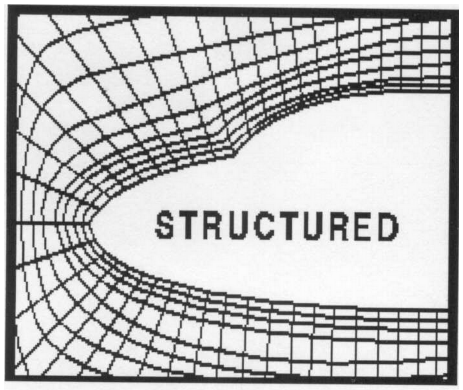
However, despite all the progress made in CFD, high-accuracy numerical simulations of complex geometries with flow conditions at high Reynolds number are still very expensive in terms of computational resources and time.

1.2 Motivation

Mathematically, the compressible Navier-Stokes equations become singular at the limit where the flow speed becomes insignificant relative to the speed of sound. At that limit the incompressible flow approximation can be used. Following the definition of incompressibility $M < 0.3$, given by [59, 182, 220], a large number of fluid problems can be considered as incompressible. The problems related to low speed aerodynamics and hydrodynamics can be classified as incompressible.

The domain of the problem needs to be split into a discrete number of elements,

denoted as mesh or grid, where the governing equations are solved. Depending on the numerical method, the grid must satisfy some requirements. For some geometries, a structured mesh may be suitable (Figure 1.5 a)). However, most of the engineering problems are associated with complex geometries, so the use of structured meshes in these problems when possible, can lead to distorted elements that could affect the accuracy of the method. One possible solution is the use of patch-based structured meshes. This solution requires a long pre-processing time during the meshing. Moreover, it is mandatory to connect the different patches that conforms the mesh. Other possible solution can be the use of unstructured meshes, Figure 1.5 b), where the pre-processing is not as demanding as patched-based meshes.

a) 

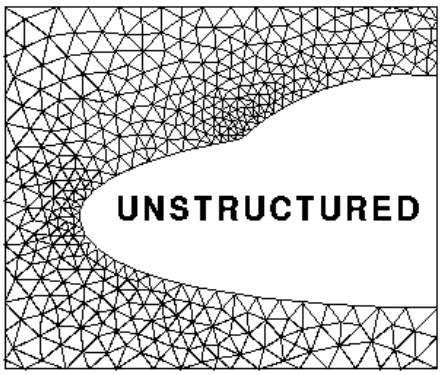
b) 

Figure 1.5: Different types of grid: a) Structured mesh and b) Unstructured mesh. From [145].

In this context, the development of very accurate numerical methods to solve the system of equations on unstructured grids is highly desirable. Since the solution of the PDE is approximated with a numerical method, there is an error of approximation between the numerical solution and the exact solution of the system of equations. One way of improving the precision in these problems is to use a finer mesh. Another alternative is to increase the order of accuracy of the method. In order to show the influence of the order and the mesh refinement, the Ringleb flow problem is simulated [25, 148, 180]. Ringleb's flow is an exact solution to the Euler equations obtained by Ringleb in 1940 [180]. It is one of the few non-trivial problems where the analytical solution of the two-dimensional Euler equations is known. This test case describes the inviscid compressible flow around a blunt obstacle [17]. The flow involves subsonic, transonic and supersonic regimes, and it is widely used as a benchmark for compressible codes. In Figure 1.6, a schematic representation of the Ringleb's flow is shown. The flow is bounded by two streamlines, ψ_{max} and ψ_{min}.

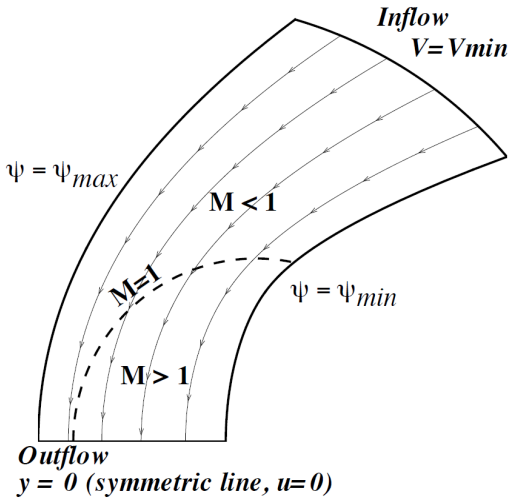


Figure 1.6: Schematic representation of the Ringleb's flow. From [130].

Ringleb obtained the exact solution using the hodograph method [180], so the physical location (x, y) is given in terms of the hodograph variables, the flow speed V and angle θ, as:

$$x = \frac{1}{\rho}\left[\frac{1}{2V^2} - \psi^2\right] + \frac{L}{2} \tag{1.6}$$

$$y = \pm\frac{\psi}{\rho V}\cos\theta \tag{1.7}$$

where the values ψ, L and ρ are defined as:

$$\psi = \frac{\sin\theta}{V} \tag{1.8}$$

$$b^2 = 1 - \frac{\gamma - 1}{2}V^2 \tag{1.9}$$

$$L = \frac{1}{b} + \frac{1}{3b^3} + \frac{1}{5b^5} - \frac{1}{2}\ln\left(\frac{1+b}{1-b}\right) \tag{1.10}$$

$$\rho = c^{\frac{2}{\gamma - 1}} \tag{1.11}$$

For this test case, only a fraction of Ringleb's domain is simulated, located at the transonic regime. The computational domain is $\Omega = [-1.15, -0.75] \times [0.15, 0.55]$. At the boundaries, the exact solution of the Ringleb's test case is imposed. The computational domain was discretized with four structured meshes of 10×10, 20×20, 40×40 and 80×80 elements. The numerical solutions are obtained using the high-order method described in [39, 148].

In Figure 1.7, the errors committed during the simulation for different orders of accuracy are plotted for the studied meshes.

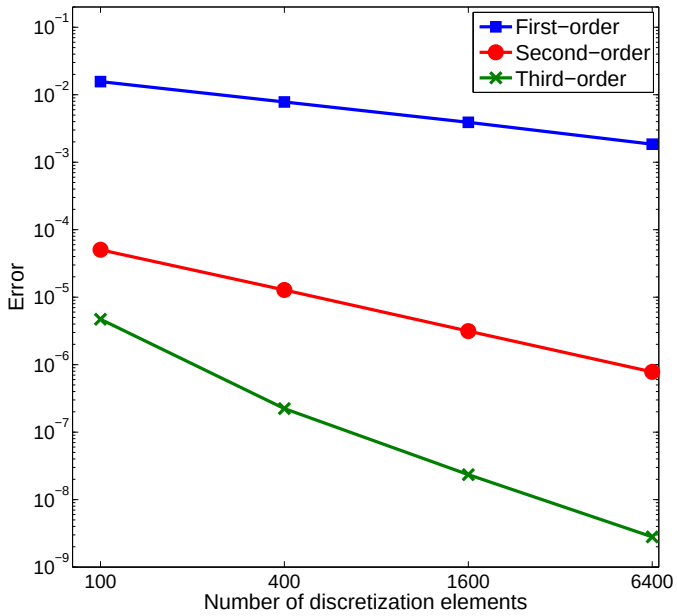


Figure 1.7: Numerical error for each mesh size with different orders of accuracy. Data taken from [39].

As Figure 1.7 shows, high-order discretization techniques offer the potential to significantly reduce the computational costs for a given accuracy by reducing the mesh size. For example, in this problem the same accuracy is obtained with a third-order method using a grid 16 times coarser than the mesh required for a second-order scheme.

1.2.1 Impact of high-order methods.

In order to demonstrate the importance of the accuracy of a numerical method, three test cases are shown:

- One-dimensional linear wave problem
- Subsonic inviscid flow around a NACA 0012
- Subsonic viscous flow around a cylinder

One-dimensional linear wave problem

In this section, the first problem presented in the First ICASE/LaRC Workshop on Benchmark Problems in Computational Aeroacoustics [71] is analyzed. The aim of this example is to show the importance of the order of accuracy for a given mesh. The following one-dimensional linear advection equation is solved

$$\frac{\partial u}{\partial t}+a\frac{\partial u}{\partial x}=0 \tag{1.12}$$

where a is the phase velocity. For simplicity, in this example, it has been taken $a=1$. The initial condition is

$$u(x,0)=0.5\ e^{[-\ln(2)(\frac{x}{3})^2]} \tag{1.13}$$

Equations (1.12) and (1.13) have the following analytic solution

$$u(x,t)=0.5\ e^{[-\ln(2)(\frac{x-at}{3})^2]} \tag{1.14}$$

In this test case, the transported wave can be obtained as a sum of harmonic waves with different frequencies and amplitudes.

The domain is $-20\leq x\leq 450$, and periodic boundary conditions are imposed at the boundaries. The domain is discretized with an uniform mesh of $\Delta x=1$. The numerical method used in this simulation is a high-order finite volume scheme, explained in detail in [149]. In Figure 1.8, the obtained numerical results for different orders of accuracy are shown at $t=400$.

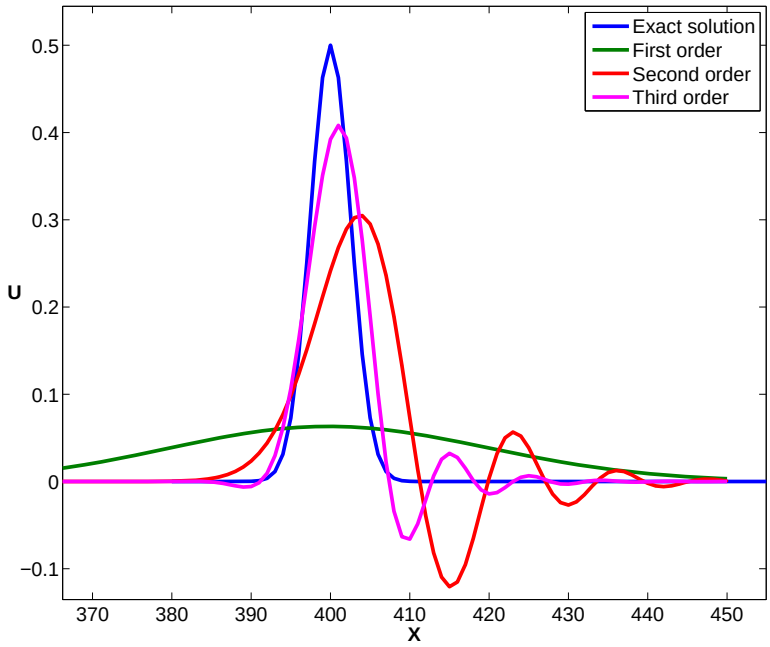


Figure 1.8: Influence of the order of resolution in the one-dimensional linear wave problem at $t = 400$.

In Figure 1.8, the obtained solutions are plotted with the exact solution computed with equation (1.14). The solution obtained with the first-order discretization has a large error in the amplitude of the wave. This is related to the dissipation error associated to low-order methods. However, the solution obtained with the second-order scheme presents less error in the amplitude of the transported wave, but spurious oscillations appear downstream the wave. These oscillations are reduced when the order is increased. These spurious oscillations are related to the dispersion error of the numerical scheme [153].

Subsonic inviscid flow around a NACA 0012

The second test case corresponds to the computation of the 2D subsonic inviscid flow around a NACA0012. In this example, the Euler equations (1.2) and (1.3) are solved in two different unstructured meshes: one of 5322 elements and other with 12246 elements. For each mesh, the solution will be computed with a first-order and a third-order finite volume method [37, 147]. The aim of this example is to compare the importance of the mesh refinement and the increase of the order of accuracy.

The freestream Mach number is $M = 0.63$ with an angle of incidence on the

NACA, also denoted angle of attack, of $\alpha = 2^{o}$.

The drag and lift coefficients, C_D and C_L, are dimensionless coefficients widely used in aerodynamics. The coefficients are computed as:

$$C_D = \frac{f_D}{\frac{1}{2}\rho\left|u_\infty\right|^2} \quad C_L = \frac{f_L}{\frac{1}{2}\rho\left|u_\infty\right|^2} \tag{1.15}$$

where the drag and lift forces, denoted as f_D and f_L, are defined as the forces exerted by the fluid to the solid. A representation of the forces exerted by the fluid to the NACA is shown in Figure 1.9. Where R denotes the total force exerted by the fluid to the profile, $\boldsymbol{f} = (f_x, f_y)$ is the force R expressed in cartesian coordinates and (f_D, f_L) is the force R expressed in the movement framework of reference. The relationship between both frameworks of reference is

$$\begin{aligned} f_D &= f_x \cos\alpha + f_y \sin\alpha \\ f_L &= -f_x \sin\alpha + f_y \cos\alpha \end{aligned} \tag{1.16}$$

where α is the angle of attack.

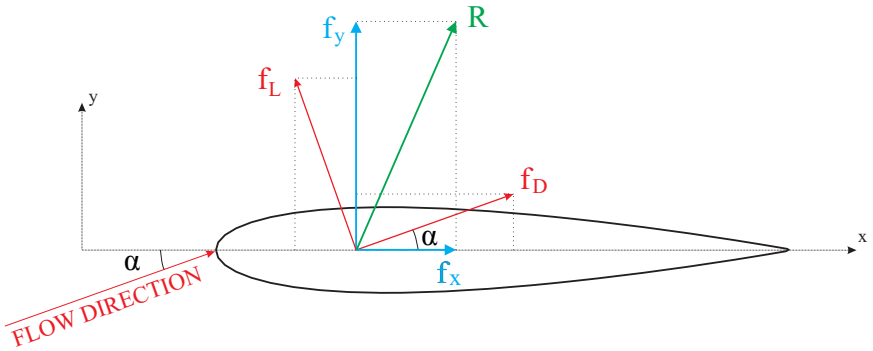


Figure 1.9: Diagram of the forces exerted on the NACA 0012.

For the flow conditions of this example, the expected lift coefficient is $C_L = 0.335$ [37, 39, 147]. Since this test case is an inviscid flow, the theoretical drag force is null, and therefore $C_D = 0$.

In Figure 1.10, the two meshes employed in this test case are presented. The coarsest mesh is denoted by mesh A, and the computational domain is discretized with 5322 quadrilateral elements. The finest mesh is denoted by mesh B and has 12246 quadrilateral elements.

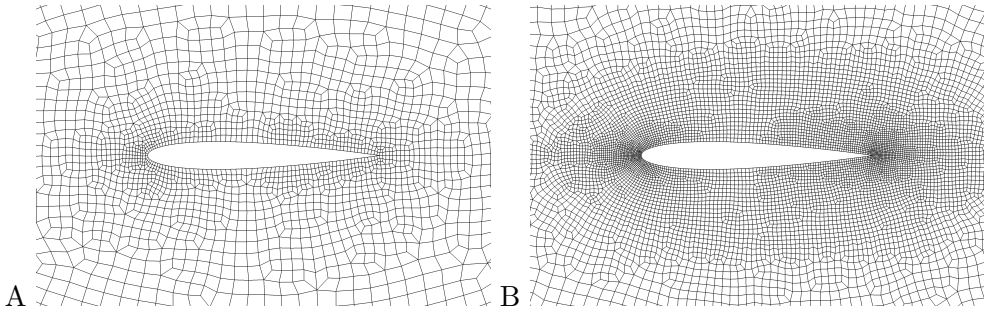


Figure 1.10: Close view of the unstructured meshes employed for the computation of the inviscid flow past a NACA0012 test case. Mesh A is the coarsest unstructured mesh with 5322 quadrilateral elements. The finest mesh is denoted by mesh B and has 12246 quadrilateral elements.

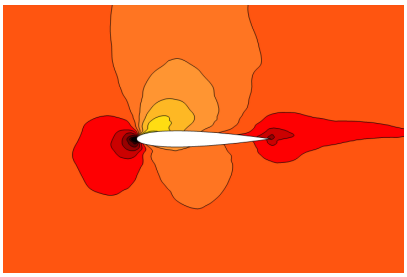
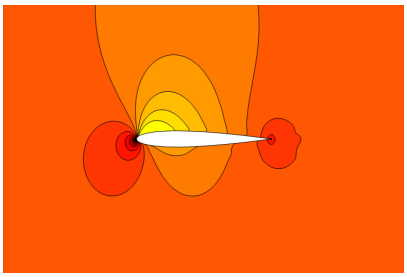
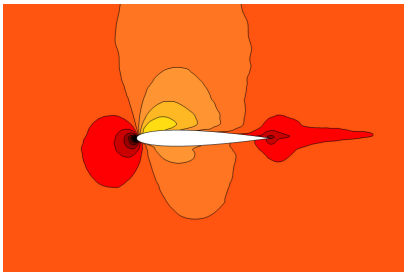
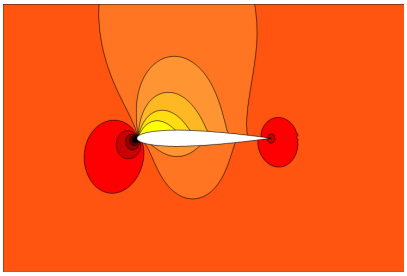
Mesh	1^{st} order discretization	3^{rd} order discretization
Mesh A (5322 elements)		
Mesh B (12246 elements)		

Figure 1.11: Mach field and contours around the NACA 0012 profile for different orders of accuracy.

Figure 1.11 represents the Mach field and contours around the NACA profile for the different orders of accuracy in both meshes. It is observed that the Mach contours computed with a first-order discretization on mesh A present a deviation in their approach to the NACA. This deviation is related to the dissipation introduced by the numerical method near the NACA. The refinement of the mesh (denoted h-refinement), also improves the solution, but for the grids employed in this test case, does not solve completely the

problem. However, increasing the order of the scheme completely solves this problem, even for the coarsest grid.

In order to quantify the error of the presented simulations, the aerodynamic coefficients C_L and C_D are compared for the different meshes and orders of accuracy in Table 1.1.

	First-order discretization		Third-order discretization	
	C_L	C_D	C_L	C_D
Mesh A (5322 elements)	0.2835	4.042E-2	0.3253	1.720E-3
Mesh B (12246 elements)	0.2947	2.594E-2	0.3264	4.018E-4
Reference solution [39]	0.3350	0	0.3350	0

Table 1.1: Comparison of the drag and lift coefficients, C_D and C_L, for the inviscid flow around a NACA 0012 for different meshes and orders of accuracy.

In Table 1.1, a comparison of the drag and lift coefficients is shown. The lift coefficient, C_L, obtained with the low-order method in mesh A presents an error of 15%. If the mesh is refined, the error in mesh B is reduced to 12%. However, the high-order discretization in mesh A presents an error of 2.9%.

As commented previously, since the Euler equations are solved (null viscosity of the fluid), the expected drag coefficient is $C_D = 0$ [39]. The error in the drag is associated to the dissipation introduced by the numerical schemes. The artificial dissipation generates numerical viscosity near the wall, that increases the entropy to the flow. In Figure 1.12, the entropy field and contours are shown and the maximum increment of entropy dS_{max} is indicated. Notice that the expected $dS_{max} = 0$.

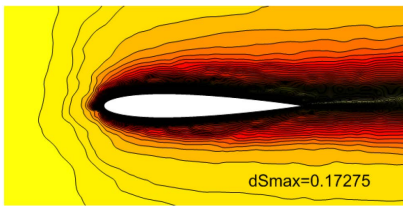
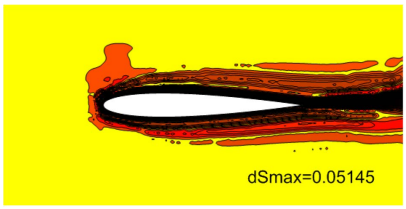
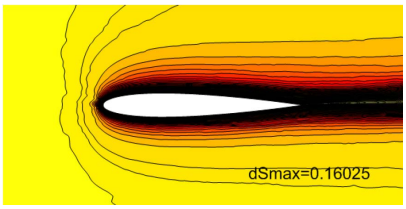
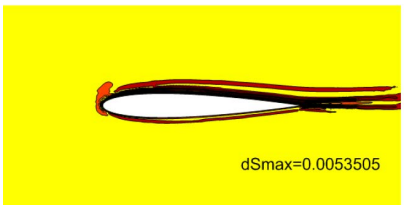
Mesh	1^{st} order discretization	3^{rd} order discretization
Mesh A (5322 elements)		
Mesh B (12246 elements)		

Figure 1.12: Entropy field and contours around the NACA 0012. The maximum increment of entropy, dS_{max}, is indicated for each case.

The solution is clearly improved when a higher-order discretization is used. Notice that the maximum increment of entropy drops one order of magnitude when the h-refinement is performed with the third-order discretization. On the other hand, the low-order solutions maintain the same order of magnitude in both meshes.

Subsonic viscous flow around a cylinder

In this section, the subsonic viscous flow around a cylinder is simulated. This test problem has been widely studied. In the 1930s, Thom got the first numerical solution of the Navier-Stokes equations for low speed viscous fluxes [201]. Kawaguti obtained in 1953 a numerical solution of the flow around a cylinder for a Reynolds number of 40.

Figure 1.4 showed the different patterns of the flow patterns past a cylinder for different Reynolds numbers. In this example, the incompressible flow around a cylinder is calculated for a Reynolds numbers of $Re = 80$. For this Reynolds number, a laminar separated and periodic flow is expected, as shown in Figure 1.13, the vortices form a periodic pattern, that is called the "Von Karman street". This repeating pattern of swirling vortices is caused by the unsteady separation of the flow of a fluid around a blunt body.

For solving this case, a structured grid was used. A close detail of this mesh is shown in figure 1.14.

For the low-order scheme, we have used a first-order discretization on pressure and momentum. In order to show the importance of the accuracy of the numerical method we

Figure 1.13: Schematic representation of the flow structure around a cylinder for a Reynolds number $5 < Re < 200$. From [84].

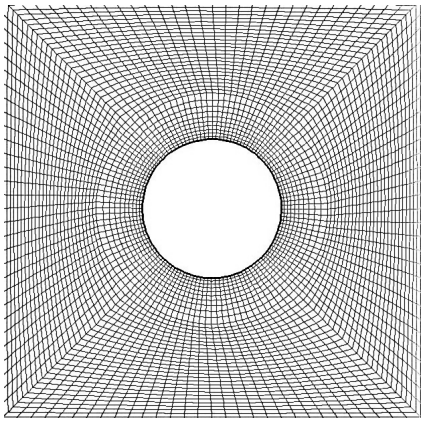


Figure 1.14: Close detail of the mesh around a cylinder

have computed a high-order solution by using third-order on momentum and second-order in pressure. Numerical results obtained for the first-order and the high-order discretization are shown in figure 1.15.

At time $t = 45$ s, the vorticity is symmetric for both cases, but each discretization obtains a different solution. At $t = 75$ s, the low-order solution is still symmetric, but the high-order solution does not.

At time $t = 150$ s it is clear that the solution obtained with the low-order scheme continuous being symmetric, but the high-order solution shows the Von Karman flow structure, as is expected in Figure 1.13 for $Re = 80$.

It is clear from Figure 1.15 that, at $t = 1000$ s, while the low-order solution is still symmetric, the solution obtained with the high-order scheme is a fully developed Von Karman street, which is the expected one.

In Figure 1.16, the time history of the lift coefficient is shown for low and high-order schemes. While the C_L obtained with the low-order discretization converges to a steady value, the lift coefficient computed with the high-order method oscillates periodically, according to the results obtained experimentally.

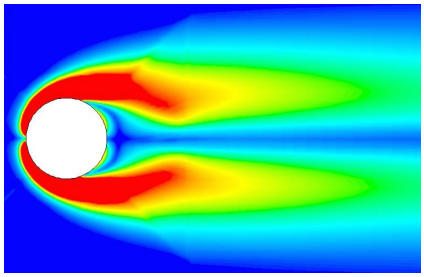
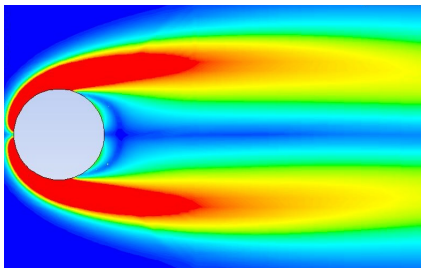
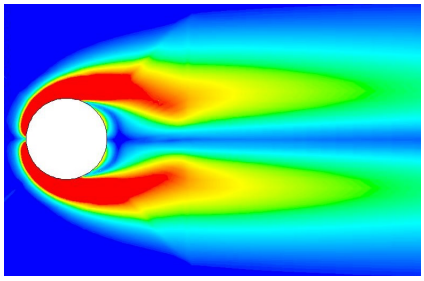
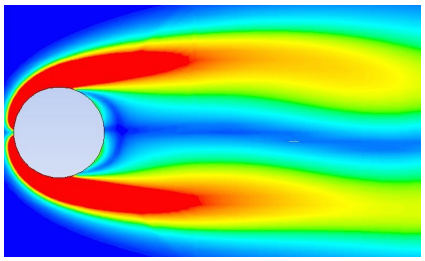
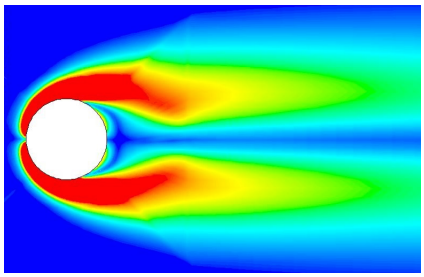
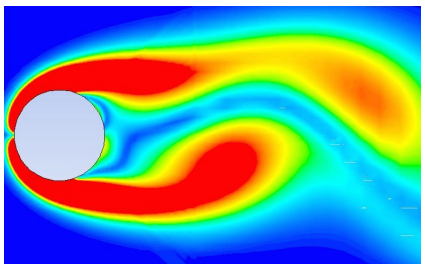
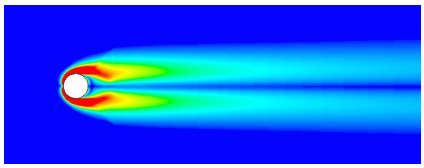
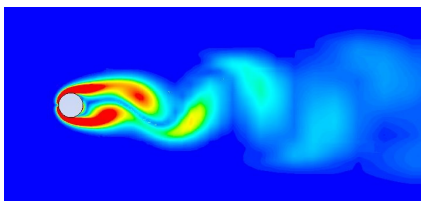
Time	Low-order discretization	High-order discretization
$t = 45$ s		
$t = 75$ s		
$t = 150$ s		
$t = 1000$ s		

Figure 1.15: Comparison of the vorticity around the cylinder for different discretization orders and time.

The explanation to these results is that the low-order scheme has a higher effective viscosity, due to the numerical dissipation introduced by the method. Since the effective Reynolds number is inversely proportional to the fluid viscosity, equation (1.5), the Reynolds number effectively computed by the numerical scheme is lower. The pattern obtained with the first-order scheme corresponds to a Reynolds number between $5 < Re < 40$. The high-order scheme obtains the right pattern expected for a $Re = 80$ flow past a cylinder.

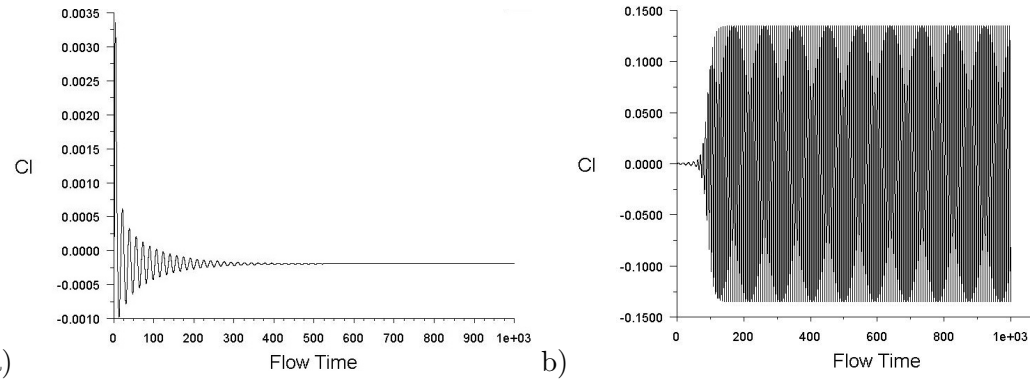


Figure 1.16: Time history of the lift coefficient for a low-order scheme, a), and high-order, b).

In the examples presented, it is remarked that a completely wrong solution is obtained if the numerical method is not accurate enough. As it is shown in Figure 1.7, high-order methods offer the possibility to reduce the approximation error with an increase of cells faster than low-order methods.

1.2.2 High-order methods for turbomachinery problems

A turbomachine is a widely used device that exchanges energy with a fluid using continuously flowing fluid and rotating blades. If the device extracts energy from the fluid it is commonly called turbine. On the other hand, if the device delivers energy to the fluid it is called compressor, fan, blower or pump depending on the fluid. Two major applications of turbomachinery include power generation and propulsion. Most of the known techniques for power generation are based on a turbomachine, which is responsible for extracting energy from a moving fluid. In power-generation, common turbomachines are the steam, gas, hydraulic, wind and tidal turbines. The steam and gas turbines are used in fossil and nuclear plants. The hydraulic turbines, also called water turbines, produces electrical power exploiting the potential energy of water stored in a reservoir. The wind and tidal turbines converts the kinetic energy from the wind and tides into electrical power.

According to [196], the power-generation throughout the world relies primarily on the steam and gas turbines for the production of electrical energy. The first practical design of a steam turbine appears in the late 19th century in the work of Parsons and de Laval and since then significant developments have been performed. Nowadays, the steam turbines are used in nuclear and combined cycle gas plants, where the nuclear energy within the atom and the chemical energy in fossil fuels are converted to thermal energy, which is transferred

to the working fluid, the steam. This thermal energy is converted to mechanical energy with the help of a high-speed steam turbine and a final conversion to electrical energy is reached with an electrical generator. A good understanding of the internal flow interaction with the turbine is the keystone of high performance turbomachinery design [222]. In steam and gas turbines, the flow regime can vary from low Mach numbers ($M < 0.3$) to supersonic flows in the exit of the nozzle [196].

The need for alternative sources of energy to satisfy our growing energy demands is increasing. The European Union (EU) has committed the target that the 20% of the final energy consumption from renewables sources by 2020. Wind and tidal energy are, according to [198], the next generation of renewable energy technologies, and they will be needed if Europe is to meet its decarbonisation targets, that committed the reduction of the greenhouse gas emissions to $80 - 95\%$ below 1990 levels by 2050 [53].

Turbomachines are the most usual device for extraction of energy from wind and tides. In Figure 1.17 three different turbines employed for renewable energy are shown.

Figure 1.17: Different turbines for renewable power generation [165]. Dashed lines denote the motion region.

The typical turbine design for wind power is the Horizontal Axis Wind Turbine (HAWT), where the rotational axis of the turbine is parallel to the flow direction. A typical HAWT is shown in Figure 1.18 a). This kind of devices has proven its efficiency on high-speed winds. However, vertical axis turbines for wind and tidal energy are currently receiving much attention from the research community to improve its efficiency. This kind of devices are capable of catching the fluid from all directions. If they are employed to obtain the energy from the wind, they are denoted as Vertical Axis Wind Turbines (VAWT) (see

a)  b)  c) 

Figure 1.18: Different tidal and wind turbines, a) Horizontal Axis Wind Turbine (HAWT) [212], b) Vertical Axis Wind Turbines (VAWT) [209], and c) Tidal Turbine [155].

Figure 1.18 b)), and if the energy is obtained from a tide, they are denoted Tidal turbines, represented in Figure 1.18 c).

The use of vertical axis turbines for the extraction of wind and tidal energy is currently under intensive research. According to [56], the typical operational fluid speed for a vertical axis turbine is shown in Table 1.2.

	Air	Sea Water
Flow speed (m/s)	≈ 12	≈ 1.5
Mach number	≈ 0.035	≈ 0.001

Table 1.2: Tidal and wind operational fluid speeds for a vertical axis turbine. Data taken from [56].

Even though the high degree of maturity achieved in the design of wind, gas and steam turbines, the efficiency of these devices may still be improved. On the other hand, current horizontal axis wind turbines design have reached the technological limit in terms of size, so new concepts of turbine are required. Moreover, the interest in tidal energy has currently increased, but current designs are in the early stages of development. High-accuracy numerical methods can help engineers to better understand the flow interaction with the turbine, which can lead to new turbine designs and to improve the efficiency of existing devices.

Numerical methods offers the potential not only for the simulation of the performance of a wind/tidal turbine, but also for the computation of the influence of wake produced by the turbomachine with the downstream turbines. Figure 1.19 shows a photo-

graph of the Horns Rev wind farm in Denmark. Notice the influence of the wake clouds with the downstream turbines.

Figure 1.19: Aerial view of the Horns Rev wind farm in Denmark. From [50].

Despite the progress made in high-order methods for CFD, common industrial simulations are usually based on second-order discretizations. These methods have been typically considered as the right choice due to their simplicity, robustness, and their effectiveness in providing a reasonably accurate solution by comparably low computational cost.

However, during the past two decades the interest in high-order methods has grown not only among the research community, but also in the field of engineering, especially in certain applications where the complex flow structure and small length scales need to be adequately resolved. In these cases, classical second-order accurate algorithms can be insufficient to accurately predict the flow in complicated geometries and complex physics. For example, the simulation of cavitation in hydrofoil or water-turbine or the computation of the noise propagation inside a fan, among many others.

As seen in Table 1.2, the operational fluid speeds for the tidal/wind vertical axis turbine results in a Mach number in the range where incompressible assumption can be applied. On the other hand, steam and gas turbines widely used for energy production present a wide range of Mach numbers, ranging from subsonic regime (where the flow can

be assumed as incompressible) to supersonic Mach numbers. Thus, the numerical method is required to deal with a broad range of flow regimes.

In order to numerically simulate the flow on a turbomachine, the relative motion of the blades needs to be taken into account. In order to preserve the accuracy of the numerical scheme, the simulation of rotating geometries needs to be of, at least, the same order than the numerical scheme. This is a crucial point in the development of high-order methods for the simulations of turbomachines, and it is addressed in this Thesis, where high-order numerical methods to simulate flows in turbomachines are developed.

1.2.3 State of art

High-order methods have been an active area of research for several decades, a brief state of art concerning high-order methods and resolution procedure of the Navier-Stokes is presented.

The Finite Difference (FD) method is one of the oldest methods to solve PDEs, since it was already known by Leonard Euler for one-dimensional cases [59]. A major disadvantage of the finite difference approach is the increase of the computational stencil for high order approaches. These large stencils are cumbersome near the edge of the domain, where no data is available. In order to solve this problem, compact finite-difference schemes were presented [114]. These schemes are a simple and powerful way to reach the objectives of high-accuracy with a low computational cost. Compared with traditional explicit finite-difference schemes, for the same order of accuracy, compact schemes are significantly more accurate with the benefit of using smaller stencils [62]. The major disadvantage is the restriction of these schemes to structured grids. Another drawback is that conservation is not enforced, unless special care is taken [139].

The Finite Element Method (FEM) was developed in the late 1950s for applications in solid mechanics [47]. In the late 1960s, Zienkiewicz and Cheung [234] applied the FEM to fluid dynamics. The computational domain is discretized in a set of non-overlapping elements. The PDE is multiplied by a test function and then integrated, leading to the so-called weak formulation. The solution is constructed using a set of basis functions. In FEM, the test and the basis functions are required to be continuous across elements. If the shape functions are equal to the test functions, the Galerkin methods are obtained. A significant contribution to CFD was the Streamline Upwind Petrov-Galerkin (SUPG), presented in the work of Heinrich et al. [76] and Hughes and Brooks [18], for convection dominated problems. The SUPG can be considered the first successful stabilization technique to prevent oscillations. The test functions are upwinded in the streamwise direction, and therefore,

the test and shape functions are different. This upwinding introduces a certain amount of diffusion only in the streamline direction. However, the SUPG is not monotone. One of the first consistent monotone methods in the finite element context is the Petrov-Galerkin FEM of Mizukami and Hughes [136]. These schemes contain modified test functions with a discontinuity-capturing term. A major drawback of the SUPG methods is the choice of the stabilization parameter, specially for unsteady problems. In this context the Taylor-Galerkin methods were presented by Donea [46]. These methods represent an attempt to take into account, by a Taylor series in time, the directional character of the propagation of information in convective transport. The time discretization is chosen so the scheme is stabilized. Another type of stabilization scheme is the Galerkin/Least-Squares (GLS) method, introduced by Hughes et al. [81]. The GLS method, a least squares form of the residual is added to the Galerkin method. These methods have been succesfully applied to the compressible Euler and Navier-Stokes [80, 189]. For the numerical resolution of the incompressible Navier-Stokes the Pressure-Stabilizing/Petrov-Galerkin (PSPG) stabilization was presented [200].

A recent approach is the Isogeometric Analysis (IGA). This approach was created to fill the gap between Computer Aided Design (CAD) and simulation [79]. This method can be seen as a generalization of FEM. Based on the isogeometric paradigm, for which the same basis functions used to represent the geometry in CAD, such as Non-Uniform Rational B-Splines (NURBS), are then used as shape functions on Finite Element Analysis framework [79]. Since the introduction of isogeometric analysis in 2005 by Hughes et al. [79], several studies have demonstrated the applicability of isogeometric analysis to problems in CFD. Some of the first studies are the work of Bazilevs et al. in 2006 [8] on the resolution of the steady-state incompressible Navier-Stokes. The application of this methodology for turbulent flows has been carried out in [9]. Recently, this method has been successfully applied to solve the compressible Euler equations [207], and complex physics such as the phase transition phenomena through the compressible Navier-Stokes Korteweg equations [66].

The Discontinuous Galerkin method (DG), introduced by Reed and Hill in 1973 [173] use piece-wise discontinuous polynomials. The DG schemes have been under development since the 1980s, and consequently these schemes have reached a certain level of maturity. Inside each element there is continuity and in order to couple the discontinuity between elements some Riemann solvers can be used, and the residual is minimized with a Galerkin approach. All of them are conservative methods at the element level, making them suitable for problems with discontinuities. The Degrees of Freedom (DoFs), on a DG framework, are either the expansion coefficients for a given set of polynomial basis functions,

or solutions at selected locations within the element. High-order DG methods can be found in the work of Bassi and Rebay [5, 6], the work of Cockburn and Shu [31], the Hybridizable Discontinuous Galerkin [33, 146], and the Compact Discontinuous Galerkin presented by Peraire and Persson [161]. A general overview of the DG method can be found in [32].

The Finite Volume (FV) method discretizes the computational domain into a set of control volumes, where the integral form of the conservation law can be applied. Finite Volume methods have become very popular in CFD based on two main advantages. First, the conservation is ensured locally in each control volume, and second, the Finite Volume method may be used on structured and unstructured meshes. Godunov in 1959, presented a conservative numerical scheme for the resolution of the non-linear hyperbolic conservation laws [65]. The main idea of the Godunov approach is that, for a given time, the solution is represented by a constant piecewise reconstruction, resulting on a first-order scheme. Bram van Leer in 1979 [213], extended Godunov's method to a second order scheme replacing the piecewise constant approximation by a linear reconstruction. The resulting method, the so-called Monotone Upstream Scheme for Conservation Laws (MUSCL) was, to the author knowledge, the first second-order finite volume scheme. In the early 1980s, Colella and Woodward [34] presented the Piecewise Parabolic Method (PPM), an extension to quadratic approximations. The main problem of the MUSCL and PPM approximations is the one-dimensional character of the approximation.

The main difficulty to achieve high-order discretizations on unstructured grids, is the need to obtain a multidimensional highly-accurate approximation of the variables. In order to obtained it, many high-order schemes have been proposed in the past, including the Essentially Non-Oscillatory (ENO) scheme introduced by Harten et al. in [73], and its variants the Weighted Essentially Non-Oscillatory (WENO) scheme [94], the Central Essentially Non-Oscillatory (CENO) scheme [88], the k-exact methods [4], the ADER approach [206] and the FV-MLS methods [36, 37], among others.

The high-order k-exact finite volume method for unstructured grids was developed by Barth et al. [4]. The basic idea lies on the use of a polynomial reconstruction for each cell and variable. The polynomial coefficients are computed so the integrals of the polynomial function over a set of control volumes recovers the average value of the variable at those elements. There is a need to compute the different polynomials approximations for each cell and variable.

The ENO schemes are high-order accurate finite-difference or finite-volume numerical methods, introduced by Harten et al. in [73] for structured grids, and improved in [1, 156, 157] for unstructured ones. The main concept of the ENO scheme is to use the smoothest stencil among others and at the same time avoid oscillations near discontinuities.

Originally, the ENO method was developed for the computation of the compressible flow at high Mach numbers. As pointed by Jiang and Shu [94], the ENO schemes have some drawbacks, such as a poor convergence rate and oscillations of the stencil where the solution is near zero. The ENO scheme was improved to the WENO schemes by Jiang and Shu [94]. In the WENO scheme, instead of using only one candidate stencil, a weighted combination of several stencils is used. The weights are defined taking into account discontinuities. For smooth regions, the scheme achieves a high-order of accuracy, while in regions near discontinuities the stencils that contain the discontinuities are assigned a nearly zero weight. Even though WENO schemes are shock-capturing schemes, they have been used to solve the incompressible Navier-Stokes as well [21]. The ENO and WENO schemes encounter convergence difficulties when selecting appropriate stencils on unstructured meshes [1]. The CENO schemes have been recently proposed by Ivan et al. in [88]. The CENO schemes attempt to deal with the computational restrictions associated with ENO and WENO. In order to avoid these restrictions, the CENO scheme uses a fixed central stencil. The main idea behind the CENO scheme is to use an hybrid solution reconstruction procedure that combines a high-order k-exact, least-squares reconstruction technique with a monotonicity preserving limited piecewise linear least-squares reconstruction algorithm [88]. The fixed central stencils are used for both the high-order k-exact reconstruction and the limited piecewise linear reconstruction.

The Arbitrary high order derivatives (ADER) method [48, 49, 206] for solving hyperbolic equations with source terms solves a generalized Riemann problem(GRP). To obtain the GRP problem, a spatial data reconstruction procedure is performed. This is made following the ENO or the WENO approach, or some other non-linear reconstruction procedure, to produce piece-wise smooth distribution of the data at a fixed time level n, typically through polynomials of arbitrary degree. In this manner at each finite volume interface one has data consisting a polynomial function on one side and another polynomial function on the opposite side. Then, the generalized Riemann problem is solved at each volume interface. The solver proposed by Toro and Titarev in [206] used a conventional, explicit time Taylor series expansion.

The FV-MLS schemes, presented in [36, 37], are a high-order accurate finite volume methods for unstructured meshes. This method is based on the Moving Least Squares (MLS) method, developed by Lancaster and Salkauskas in 1981 [111]. MLS is used to obtain the successive derivatives needed for the high-order reconstruction of variables using Taylor series. The numerical method has been tested for the resolution of the compressible Euler and Navier-Stokes equations [37, 147]. The FV-MLS has also been applied to the numerical resolution of the Linearized Euler Equations for aeroacoustics [153], and to the

shallow water equations [38].

The Spectral methods, like the Spectral Finite Volume (SV) [224] and the Spectral Difference scheme (SD) [124], employ a piecewise-polynomial reconstruction inside each cell. As an alternative to the k-exact high-order scheme, the SV method was developed by Wang et al. [224] to improve the efficiency. In order to achieve it, each cell (denoted by spectral volume) is partitioned in a geometrically similar manner into a a subset of cells, called control volumes. Using this approach the same polynomial reconstruction can be applied for every spectral volume. Inside each spectral volume there is a continuous reconstruction between the control volumes and the discontinuity arises between spectral volumes. These methods share some similarities with the Discontinuous Galerkin method. The difference between the SV and the SD resides that the SV methods are based on the integral form of the conservative equation, and the SD approach on the differential form. In a SV method, the Degrees of Freedom (DoFs) are sub-cell averaged solutions, while in the SD method, the DoFs are the solutions at the points. These methods are well suited for simulations on unstructured grids while applying ENO/WENO/CENO may become cumbersome for unstructured grids. However, it has been shown that the family spectral high-order methods also suffers from Gibbs phenomenon when discontinuities are present [26].

On a finite volume framework, two main families have been developed to compute either compressible or incompressible flows: "Density-based" and "pressure-based" solvers. Density-based solvers [34, 181, 213] are used for the computation of flows when compressibility effects are important (mainly transonic, supersonic and hypersonic flows), whereas pressure-based solvers [29, 159, 164] are designed to compute incompressible flows. In both techniques, the velocity is obtained from the momentum equations. The difference between the two approaches is the computation of the pressure field. In density-based solvers, the density is computed from the continuity equation and then the pressure is obtained from an equation of state. On the other hand, in the pressure-based approach, the pressure field is extracted by solving a pressure or pressure correction equation which is obtained by manipulating continuity and momentum equations.

As it is shown in Figure 1.3, the incompressibility limit is placed by several authors [59, 182, 220] at $M < 0.3$. From a practical point of view, density-based solvers are not suitable for flows with Mach number below that limit [182]. These solvers present a number of problems: stiffness of the equations, cancellation in the pressure variable [187] and the loss of accuracy due to a wrong numerical diffusion (accuracy problem). As pointed in [59], due the stiffness of the equations at the incompressibility limit, the necessary time step in an explicit method needs to be very small. The reason is that the time step is determined by the larger of the two velocities that the information travels: the velocity of the flow and

the speed of sound. Therefore, in the low speed limit, the required time step may be much smaller that the time step required for a incompressible solver.

The accuracy problem of density-based solvers in low-Mach flows is originated by the introduction of spurious pressure and velocity waves that avoid the velocity field to verify the zero-divergence constraint [43, 69, 70, 105, 116]. Preconditioning techniques [28, 69, 70, 208, 215] have been developed to increase the accuracy (and also to overcome the stiffness problem) of Godunov schemes in low Mach flows. The preconditioning matrix multiplies the time derivatives of the set of equations, with the effect of re-scaling the eigenvalues (acoustic-speed) of the system, but paying the price of spoiling the temporal accuracy of the scheme. Thus, this technique was initially developed for steady flows, but extensions to unsteady flow have been proposed by using a dual time stepping technique [132, 219]. Recently, it was reported [177] that the accuracy problem is not observed when the Roe's approximate Riemann solver is used in a first-order finite volume scheme on triangular grids. This behavior is related to cancellation of the normal velocity jump. Based on this finding, a fix for the numerical flux of Roe is presented in [179]. This fix reduces the jumps in the normal velocity component of the Roe's Riemann solver. Other fixes have also been proposed for Roe flux [203] and for other Riemann solvers, such as the HLL-family [117]. To the author knowledge, all these fixes have been applied to second-order methods, at most. Several flux-splitting type schemes accurate at low and also high Mach have been presented in [120, 190, 191]. On the other hand, some authors have reported Discontinuous Galerkin (DG) solutions of low-Mach flow with and without preconditioning [7, 55, 127]. The high-order approach reduces the jumps in the normal velocity component alleviating the accuracy problem. It has been shown, however, that in DG schemes the accuracy problem is not completely solved [7], since a grid dependency with the Mach number still remains.

For the numerical resolution of the incompressible Navier-Stokes, the main problem is the difficulty in coupling changes of the velocity field with changes in the pressure field while satisfying the continuity equation. The reason is the absence of a transient term in the continuity equation, so the equations are decoupled and the continuity equation can be considered as a divergence-free constraint that the velocity field is required to satisfy. In order to overcome this problem, a pressure-based method has been generally used. In this approach, the pressure is used as a mapping parameter to satisfy the continuity equation. The SIMPLE and related algorithms [159] have been widely used to compute incompressible flows, and several authors have developed second-order methods to extend these algorithms to all-speed flows [101, 141]. Other approaches to solve the incompressible Navier-Stokes are the artificial compressibility method and methods based on derived variables. The method

of artificial compressibility proposed by Chorin [29], introduces a pseudo-time derivative of the density into the continuity equation, and therefore a density-solver can be used. The original form of Chorin's method was developed for steady state problems, and Peyret and Taylor [164] extended it to a time accurate formulation. Later, the method was fully extended to general three dimensions by Kwak et al. [108].

In order to remove the pressure from the formulation, different approaches introducing other variables instead have been developed. The most common approach of this kind is the stream function vorticity method [109]. The extension of this approach to 3D problems adds much complexity to the formulation and is more expensive that methods which solve velocity and pressure [109].

For the numerical simulation of the flow on a turbomachine, the relative motion of the blades needs to be taken into account. Several techniques have been developed to treat problems with moving boundaries: the Immersed Boundary methods [163], Chimera methods [195] and body fitted techniques, among others. Commonly used body-fitted techniques for the simulation of turbomachines are the Multiple Reference Frames [126] (MRF) and Sliding Mesh (SM) models [131]. In these models, the domain is divided into two regions, an inner region containing the rotor (moving) and an outer region containing the stator (fixes). For the MRF model, steady-state calculations are performed with a rotating reference frame in the rotor region and a stationary reference frame in the stator region. This method simulates steady-state cases, so it neglects the unsteady flow interactions, such as wakes, potential and shock interactions between different blades that conform the turbine. Numerical simulations of turbomachinery using MRF can be found in [19, 45]. In Sliding Mesh methods, the rotor is allowed to slide relative to the stator in discrete time steps and thus, time-dependent calculations can be performed. As it is time dependent, the SM method is the more accurate representation of the actual phenomenon of the impeller rotation. However, it is computationally demanding. Other body-fitted approach are Adaptive methods [77], which are based on the deformation of the elements when the blades are moving. These techniques are efficient for small body motions, but for large motions there is a need for remeshing, which could be unfeasible for practical computations. A recent application of an Adaptive method to the numerical simulation of a turbine can be found in [78].

The Immersed Boundary (IB) method was introduced in 1972 by Peskin as a way to simulate the beating human heart [163]. In the classical IB, the body boundary conditions do not rely on the shape of the mesh, they are introduced as forcing terms in the governing equations. These methods are attractive because of their simplicity, but their major drawbacks are the occurrence of non-divergence free velocities in incompressible

flows, spurious non physical pressure oscillations in compressible flows, and the difficulty to get high-order schemes near the wall [97]. A second class of IB methods is the cut-cell method introduced by Clarke [30]. In the cut-cell method the immersed boundaries cut the mesh, creating a set of irregularly shaped cells upon which the equations are discretized. This method adds complexity, since each time step the original mesh needs to be cut by the moving bodies, the geometrical information and the integration points need to be recomputed. Application of IB methods to turbomachinery can be found in [98, 232].

Chimera methods, which are sometimes also referred to as overlapping or overset meshes, is a mesh strategy first used by Steger et al. in 1983 [195]. The main advantages of this method are its ability to handle the complex geometries and the relative motion of bodies in dynamic simulations. The flexible discretization offered by Chimera methods relies on the fact that each part of the computational geometry can be meshed independently, resulting on a series of grids that overlap each other to form a single computational domain. The set of grids periodically update and exchange boundary information through a proper interpolation. However, the use of nonconservative interpolation approaches to update variables in the overlapped region can give rise to spurious solutions. The Chimera methods has been used in the numerical simulation of turbines in [118, 231].

Francois et al. [60] showed a comparative study between Chimera and Sliding Mesh techniques for unsteady simulations of counter rotating open-rotors. They conclude that Sliding Mesh techniques give similar accuracy but the later requires much less memory than the Chimera approach. Chimera methods requires an interpolation scheme to couple the overlapping meshes, leading in many cases to non-conservative methods.

Nowadays, sliding mesh techniques are commonly used to compute non-axisymmetrical unsteady flow fields and corresponding aerodynamic performances of cross-flow fans [138, 204] and wind turbines [10, 89, 91, 92, 95, 99, 100, 113]. Note also that the sliding mesh algorithm was used by Steijl and Barakos [194] for the computational fluid dynamic analysis of helicopter rotor-fuselage aerodynamics.

However, most of the sliding mesh methods proposed in the literature belong to the family of first or second order interpolation schemes. Therefore, they can not be used in conjunction with higher-order numerical schemes without depreciating the overall accuracy of the numerical methods. To the author knowledge, few studies addressed such problem. One of them is a high order ($order \geq 3$) h/p Discontinuous Galerkin method with sliding mesh capabilities, recently proposed by Ferrer and Willden [58] for the computation of the unsteady incompressible flow field of a three bladed cross-flow turbine.

1.3 Objectives

The present thesis continues the research developed by the Group of Numerical Methods In Engineering (GMNI) of the *Universidade da Coruña* (UDC) in the application of meshless techniques to the development of high-order finite volume methods for Computational Fluid Dynamics (CFD).

In 2005, the GMNI introduced the use of Moving Least Squares to obtain high-order finite volume methods, leading to the FV-MLS method [36, 37]. The resulting scheme was applied to the numerical resolution of the compressible Navier-Stokes. Since then, the FV-MLS has been applied to solve the linearized Euler equations for aeroacoustics [153], compressible flows at high Mach number [151], shallow water dynamics [38], the resolution of the Cahn-Hilliard and the Kuramoto–Sivashinsky equations [41], and turbulent flows [152] among others. The FV-MLS allows to achieve high spatial orders of accuracy on unstructured grids without increasing the number of degrees of freedom.

The great geometrical flexibility offered by unstructured grids makes them highly effective for dealing with complex geometries, that are more typical of practical engineering applications. Although second-order accurate methods are currently the standard in industry for computations on unstructured grids, high-order discretization techniques offer the potential to significantly reduce the computational cost for a given accuracy by reducing the mesh size.

While the resolution of compressible flows with the FV-MLS have been extensively studied [36, 37, 39, 40, 147, 153, 169], the application of the FV-MLS method to incompressible or nearly incompressible flows has never been addressed with this methodology.

The simulation of rotating geometries, such as turbomachines, requires a method to "couple" the different mesh regions without losing accuracy, and at the same time allows the relative mesh motion. In order to preserve the accuracy of the scheme, the coupling between different grids needs to be of, at least, the same order than the numerical scheme.

The primary objective of this thesis is the development, analysis and validation of novel and highly accurate numerical methods to simulate turbomachinery flow on unstructured meshes. In order to achieve it, several ingredients are needed:

1. The development and verification of a high-order finite volume scheme for the resolution of the Navier-Stokes equations on complex geometries for all-speed flows, that is a wide range of Mach numbers in the solution, from nearly incompressible to supersonic flows.

2. The development and analysis of a novel high-order pressure-based finite volume

scheme for the resolution of the incompressible Navier-Stokes equations on complex geometries. Even though all fluids are compressible in an absolute sense, the incompressible flow approximation can be assumed when the flow speed is small enough compared with the speed of sound of the medium. At this speed, the resolution of the compressible Navier-Stokes with an all-speed scheme is not practical, motivated by a slow convergence to the solution due the large disparity between the acoustic and the flow speed.

3. The development and analysis of new high-order Sliding mesh methods in a finite volume framework. The new methods will be verified in several compressible and incompressible flow test cases.

Some secondary objectives are:

- Studying the order of convergence of the presented numerical methods.
- Comparing the numerical results obtained with reference values in order to check the accuracy of the presented methodologies.

1.4 Overview of the thesis

The document is organized in five chapters. Chapter 2 introduces the basic ingredients of this thesis, that is, the finite volume framework, the meshless technique Moving Least Squares (MLS), and finally the FV-MLS, proposed in [37]. In Chapter 3, the FV-MLS is applied on a density-based solver for the numerical resolution of the compressible Navier-Stokes for all-speed flow. The numerical method is analyzed in several steady and unsteady test cases. In Chapter 4, the FV-MLS method is extended to the resolution of the incompressible Navier-Stokes equations. In order to achieve it, a high-order pressure-based formulation is presented. The accuracy of the scheme is studied on typical benchmarks for steady and unsteady cases. Chapter 5 is devoted to the development of high-order sliding-mesh techniques for a finite volume framework. The resulting schemes are tested in compressible and incompressible flows. To conclude Chapter 5, a cross-flow turbine is analyzed. Finally, conclusions, and directions for future research are summarized in Chapter 6.

Chapter 2

Finite Volume, and Moving Least Squares approximants. The FV-MLS method.

2.1 Introduction

The finite volume method is a discretization technique for partial differential equations (PDEs) that is well suited for the numerical simulation of conservations laws. This discretization technique has been extensively used for several applications such as fluid mechanics in engineering and heat and mass transfer.

In a finite volume framework, a mesh is created for a given spatial domain Ω_T. For some geometries, a structured mesh may be suitable. However, most of the engineering problems are associated with complex geometries, so the use of structured meshes in these problems when possible, can lead to distorted elements that could affect the accuracy of the method. One possible solution is the use of patch-based structured meshes. This solution requires a long pre-processing time during the meshing. Moreover, it is mandatory to connect the different patches that conforms the mesh. Other possible solution can be the use of unstructured meshes, where the pre-processing is not as demanding as patched-based meshes.

The finite volume technique may be used on arbitrary geometries (structured or unstructured meshes). In many engineering problems, the complexity of the associated geometry make the use of unstructured meshes desirable. In unstructured meshes, most of the schemes commonly employed in industry are second-order accurate.

In this context, the FV-MLS method is a finite volume method based on a mesh-

less technique called Moving Least Squares (MLS). The basic idea of the FV-MLS method is to use the Moving Least Squares approximants to reconstruct the variables and their derivatives from the point values of a series of arbitrary distributed particles. This reconstruction of the function and the calculation of derivatives is necessary in the finite volume high-order discretization.

In this chapter the fundamentals of the finite volume method are briefly presented. Next, Moving Least Squares approximants are exposed. Finally, the FV-MLS method is introduced.

2.2 The Finite Volume Method

Let us consider a generic conservation law for the two dimensional domain Ω_T under the general form

$$\partial_t u + \nabla \cdot \boldsymbol{\mathcal{F}} = S \qquad \text{in} \quad \Omega_T \tag{2.1}$$

where u is the conservative variable of $(\boldsymbol{x}, t) \in \Omega \times (0, T)$, $\boldsymbol{\mathcal{F}}$ is the flux vector and S is a source term. The conservation law needs to be supplemented with some suitable initial condition of the form $u(\boldsymbol{x}, 0) = u_{ini}(\boldsymbol{x})$, as well as some boundary conditions, which depend on the considered equation.

As a starting point, the problem domain Ω_T needs to be decomposed into a finite number of subdomains, $\Omega_i|_{i=,\dots,N}$, called control volumes where the variable is stored. The definition of a given control volume is arbitrary. Thus, we can use the cell-centered approach, where the variable is stored at cell centroid, as shown in Figure 2.1 a), or a cell-vertex finite volume, which stores the variable at the nodes , Figure 2.1 b).

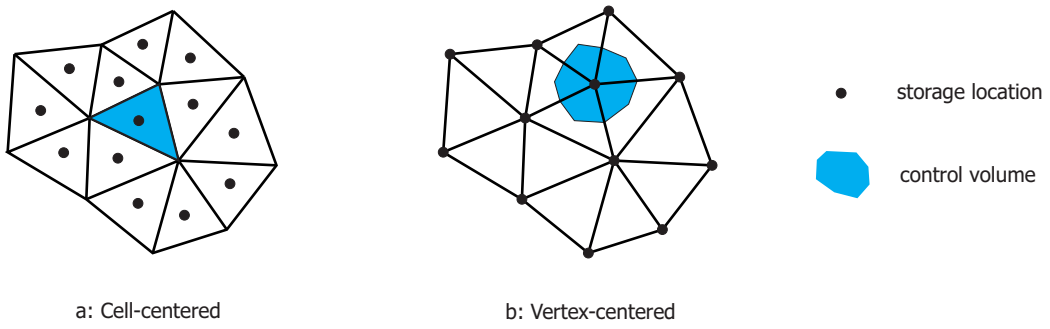


Figure 2.1: Control volume definitions for finite volume methods: a) cell-centered and b) vertex-centered approach.

Finite volume methods locally balance equation (2.1) for each control volume. In

order to achieve it, equation (2.1) needs to be rewritten into the integral balance for each control volume. That is, for the control volume I the conservation law reads in integral form as

$$\int_{\Omega_I} \partial_t u \, d\Omega + \int_{\Omega_I} \nabla \cdot \boldsymbol{\mathcal{F}} \, d\Omega = \int_{\Omega_I} S \, d\Omega \tag{2.2}$$

Applying the divergence theorem to the flux term of equation (2.2), the following equation is obtained:

$$\int_{\Omega_I} \partial_t u \, d\Omega + \int_{\Gamma_I} \boldsymbol{\mathcal{F}} \cdot \boldsymbol{n} \, d\Gamma = \int_{\Omega_I} S \, d\Omega \tag{2.3}$$

where Ω_I is the control volume area, Γ_I is the control volume perimeter and $\boldsymbol{n} = (n_x, n_y)^T$ is the unitary exterior normal of the contour, as seen in Figure 2.2.

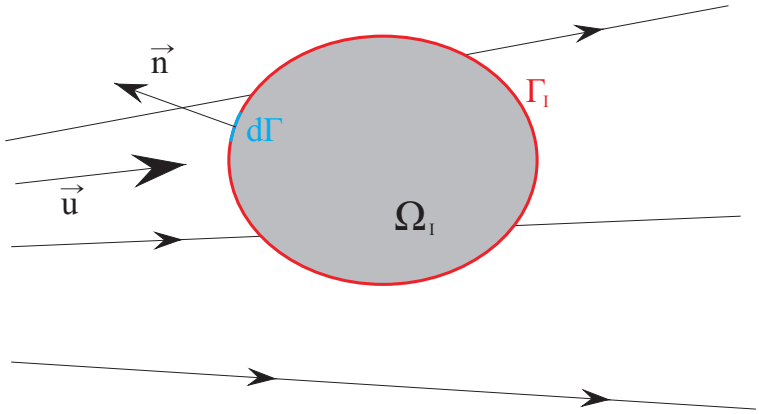


Figure 2.2: Definition of the control volume Ω_I.

The temporal integral term of equation (2.3) is solved as

$$\int_{\Omega_I} \frac{\partial u}{\partial t} d\Omega = \Omega_I \frac{\partial u_I}{\partial t} \tag{2.4}$$

where u_I is the average value of the conservative variable over the control volume I, that is

$$u_I = \frac{1}{\Omega_I} \int_{\Omega_I} u \, d\Omega \tag{2.5}$$

Applying Gauss quadrature to evaluate the flux integral in equation (2.3) produces the following expression

$$\int_{\Gamma_I} \boldsymbol{\mathcal{F}} \cdot \boldsymbol{n} \, d\Gamma = \sum_{j=1}^{N_f} \sum_{ig=1}^{N_G} \left[\boldsymbol{\mathcal{F}}_j \cdot \hat{\boldsymbol{n}}_j \right]_{ig} \mathcal{W}_{ig} \tag{2.6}$$

where N_f is the number of faces (3D) or edges (2D) of the control volume, N_G represents the number of quadrature points, $\mathcal{W}_{ig}$ is the corresponding quadrature weight for the quadrature point at cell edges, denoted with subscript ig, and $\hat{\boldsymbol{n}}_j$ is the unitary normal $\boldsymbol{n}$ times the length of the edge j.

The generic conservation law can be rewritten as

$$\Omega_I \frac{\partial u_I}{\partial t} + \sum_{j=1}^{N_f} \sum_{ig=1}^{N_G} \left[\boldsymbol{\mathcal{F}}_j \cdot \hat{\boldsymbol{n}}_j \right]_{ig} \mathcal{W}_{ig} = \int_{\Omega_I} S \, d\Omega \tag{2.7}$$

The accuracy and order of the numerical method will be determined by the computations of each term in equation (2.7). Note that the flux vector $\boldsymbol{\mathcal{F}}$ must be evaluated at the integration points located at each cell edge, denoted by "**x**" in Figure 2.3. The quadrature used to evaluate the integrals should be at least of the same order as the reconstruction, so the number of integration points will depend on the selected quadrature order.

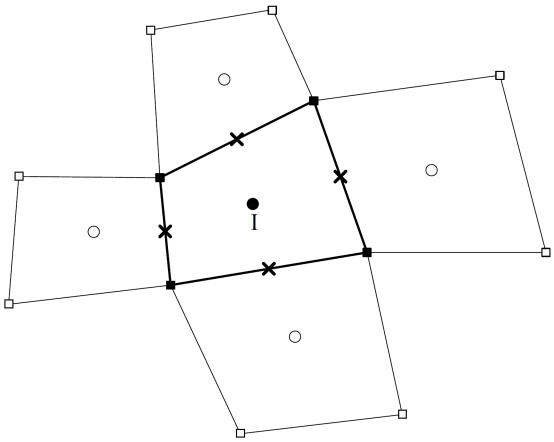


Figure 2.3: Schematic illustration of the cell I and its integration points ("**x**").

The flux, $\boldsymbol{\mathcal{F}}$, of equation (2.7), can be split into a hyperbolic-like part, $\boldsymbol{\mathcal{F}}^{\mathcal{H}}$, and a elliptic-like part, $\boldsymbol{\mathcal{F}}^{\mathcal{V}}$, as suggested in [39]

$$\Omega_I \frac{\partial u_I}{\partial t} + \sum_{j=1}^{N_f} \sum_{ig=1}^{N_G} \left[\left(\boldsymbol{\mathcal{F}}^{\mathcal{H}}{}_j - \boldsymbol{\mathcal{F}}^{\mathcal{V}}{}_j \right) \cdot \hat{\boldsymbol{n}}_j \right]_{ig} \mathcal{W}_{ig} = \int_{\Omega_I} S \, d\Omega \tag{2.8}$$

Godunov in 1959, presented a conservative numerical scheme for the resolution of the non-linear hyperbolic conservation laws [65]. The numerical scheme was based on the exact solution of a Riemann problem. The main idea of the Godunov approach is that, for a given time, the solution is represented by a constant piecewise reconstruction

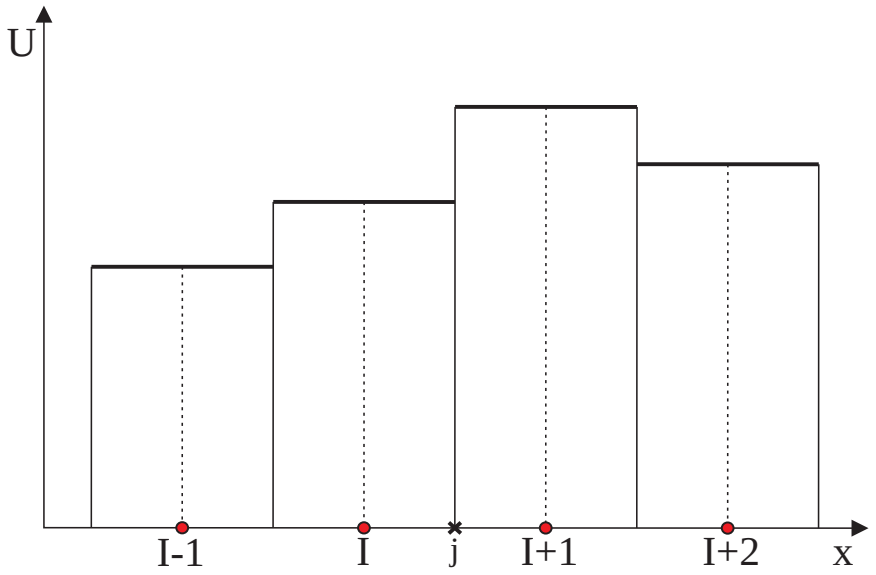


Figure 2.4: Constant piecewise solution.

,as seen in Figure 2.4. Therefore, the evaluation of the hyperbolic flux, $\boldsymbol{\mathcal{F}^{\mathcal{H}}}$ in equation (2.8), does not have a unique solution, since there is a discontinuity at cell edges $\boldsymbol{\mathcal{F}^{\mathcal{H}}}(u^+) \neq \boldsymbol{\mathcal{F}^{\mathcal{H}}}(u^-)$. In order to circumvent this problem, Godunov's original method exactly solves a local Riemann problem with the two states (u^+, u^-). Since the solution of a Riemann problem requires to solve a non-linear system of equations, several approximations have been presented [181, 183]. The approximation, also called numerical flux, obtains a single value of the hyperbolic flux $\boldsymbol{\mathcal{F}^{\mathcal{H}}}$ from both states of the variable at the interface.

Van Leer in [213], extended Godunov's method to a second-order scheme replacing the piecewise constant approximation, Figure 2.4, by a linear reconstruction. This statement is represented in Figure 2.5. Thus, one possible to increase the accuracy of the numerical scheme is to improve the reconstruction of the variable locally in cell I.

The hyperbolic-like part, $\boldsymbol{\mathcal{F}^{\mathcal{H}}}$, is computed at the integration point using a "broken" reconstruction. This high-order reconstruction is achieved using a high-order Taylor series from the cell centroid. Mathematically, this statement can be expressed as $\boldsymbol{\mathcal{F}^{\mathcal{H}}} = \boldsymbol{\mathcal{H}}(u^+, u^-)$, where $\boldsymbol{\mathcal{H}}$ is a numerical flux that solves the Riemann problem formed at the integration point. The reconstructed values u^+ and u^- are the left and right states of cell I. In Figure 2.5 the "broken" reconstruction is shown for a linear Taylor reconstruction.

In general, the order of accuracy of the "broken" reconstruction is required to be the same as the original continuous reconstruction. The cell-wise "broken" reconstruction defined for the hyperbolic fluxes, showed in Figure 2.5, is obtained using Taylor series

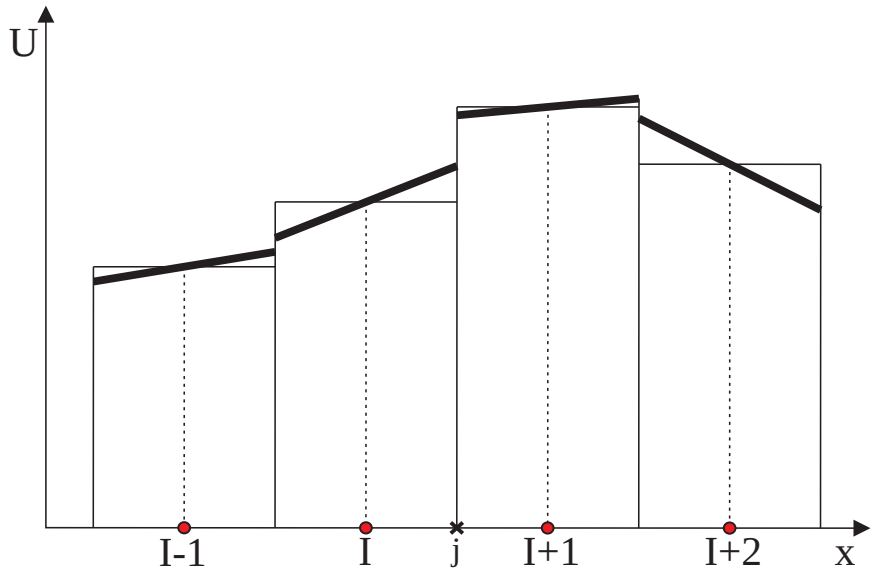


Figure 2.5: Piecewise linear reconstruction inside the cells. Notice the discontinuity across the interface j.

expansion. For example, a quadratic reconstruction inside cell I, reads

$$u(\boldsymbol{x})\Big|_{\boldsymbol{x}=\boldsymbol{x}_j} = u_I + \boldsymbol{\nabla} u_I \cdot (\boldsymbol{x}_j - \boldsymbol{x}_I) + \frac{1}{2} (\boldsymbol{x}_j - \boldsymbol{x}_I)^T \boldsymbol{H} \ (\boldsymbol{x}_j - \boldsymbol{x}_I) \tag{2.9}$$

where the gradient $\boldsymbol{\nabla u}_I$ and the Hessian matrix $\boldsymbol{H}$ involve the successive derivatives computed at cell centroids.

As suggested in [37, 147], for an unsteady problem a correction needs to be added to equation (2.9) in order to preserve the mean value of the cell I. Note that the storage value at cell centroid is the average value of the variable, as stated in equation (2.5).

$$\begin{aligned} u(\boldsymbol{x}) = u_I + \boldsymbol{\nabla} u_I \cdot (\boldsymbol{x} - \boldsymbol{x}_I) + \frac{1}{2} (\boldsymbol{x} - \boldsymbol{x}_I)^T \boldsymbol{H} \ (\boldsymbol{x} - \boldsymbol{x}_I) \\ - \frac{1}{2\ \Omega_I} \left(I_{xx} \frac{\partial^2 u}{\partial x^2} + 2 I_{xy} \frac{\partial^2 u}{\partial x \partial y} + I_{yy} \frac{\partial^2 u}{\partial y^2} \right) \end{aligned} \tag{2.10}$$

where

$$I_{xx} = \int_{\Omega_I} (x - x_I)^2 \ d\Omega, \quad I_{yy} = \int_{\Omega_I} (y - y_I)^2 \ d\Omega, \quad I_{xy} = \int_{\Omega_I} (x - x_I)(y - y_I) \ d\Omega \tag{2.11}$$

Note that the introduction of the correction terms does not reduce the order of the approximation given by equation (2.11). For a stationary case the correction terms are not needed, therefore the reconstruction defined in equation (2.9) can be used.

The elliptic-like fluxes are computed centered at the integration points. That is, in Figure 2.3 the elliptic-like flux, $\boldsymbol{\mathcal{F}^{\mathcal{V}}}$, will be computed directly at the integration point.

The difficulty to achieve a high-order method resides on the computation of the derivatives needed for the Taylor reconstruction and for the evaluation of the viscous fluxes. In the following section a meshless technique is presented circumvent the computation of the high-order reconstruction.

2.3 The Moving Least Squares method

In this section the Moving Least Squares (MLS) is presented. This method was originally devised by Lancaster and Salkauskas in 1981 for data processing and surface generation [111]. The MLS method has become very popular among the meshless community. In 1992, Nayroles et al. employed the MLS method to obtain the shape functions of the Diffuse Element Method (DLM) [144]. Belystschko combined in 1994 the MLS approach with the Galerkin method, to develop the Element-Free Galerkin (EFG) method [12].

This class of approximation methods is particularly well suited for the reconstruction of a given function and its successive derivatives from scattered, pointwise data. An additional advantage of this approach is the multidimensional nature of the computation of the derivatives.

Let $u(\boldsymbol{x})$ be the function of the variable defined in the domain Ω_T. Moving Least Squares (MLS) approximates $u(\boldsymbol{x})$ at a given point $\boldsymbol{x}$ with a function $\widehat{u}(\boldsymbol{x})$ obtained through a weighted least squares fitting of $u(\boldsymbol{x})$ in a compact $\Omega_{\boldsymbol{x}}$, as

$$u(\boldsymbol{x}) \approx \widehat{u}(\boldsymbol{x}) = \sum_{i=1}^{m} p_i(\boldsymbol{x})\alpha_i(\boldsymbol{z})\Big|_{\boldsymbol{z}=\boldsymbol{x}} = \boldsymbol{p}^T(\boldsymbol{x})\boldsymbol{\alpha}(\boldsymbol{z})\Big|_{\boldsymbol{z}=\boldsymbol{x}} \tag{2.12}$$

where $\boldsymbol{p}^T(\boldsymbol{x})$ is an m dimensional basis of functions. That basis is usually formed by polynomials and for the one dimensional case it can be defined as

$$\boldsymbol{p}(x) = (1, x, x^2, \ldots, x^{m-1}) \tag{2.13}$$

and $\boldsymbol{\alpha}(\boldsymbol{z})\Big|_{\boldsymbol{z}=\boldsymbol{x}}$ is a vector of m coefficients to be determined. Note that the coefficients are function of $\boldsymbol{x}$. In order to obtain the values, a functional of weighted residual is constructed using the approximated field function $\widehat{u}(\boldsymbol{x})$ and the field function $u(\boldsymbol{x})$ at the compact support $\Omega_{\boldsymbol{x}}$.

$$J(\boldsymbol{\alpha}(\boldsymbol{z})\Big|_{\boldsymbol{z}=\boldsymbol{x}}) = \int_{\boldsymbol{y}\in\Omega_{\mathbf{x}}} W(\boldsymbol{z}-\boldsymbol{y}, h)\Big|_{\boldsymbol{z}=\boldsymbol{x}} \left[u(\boldsymbol{y}) - \boldsymbol{p}^T(\boldsymbol{y})\boldsymbol{\alpha}(\boldsymbol{z})\Big|_{\boldsymbol{z}=\boldsymbol{x}}\right]^2 d\Omega_{\boldsymbol{x}} \tag{2.14}$$

where $W(\boldsymbol{z}-\boldsymbol{y},h)\Big|_{\boldsymbol{z}=\boldsymbol{x}}$ is a weighted function (also known as *kernel*) centered at $\boldsymbol{z}=\boldsymbol{x}$. The parameter h is the smoothing length that defines the compact support ($\Omega_{\boldsymbol{x}}$) of the *kernel*.

The minimization of the functional J with the parameters $\boldsymbol{\alpha}$ leads to the following equation:

$$\int_{\boldsymbol{y}\in\Omega_{\boldsymbol{x}}} \boldsymbol{p}(\boldsymbol{y})W(\boldsymbol{z}-\boldsymbol{y},h)\Big|_{\boldsymbol{z}=\boldsymbol{x}}\boldsymbol{u}(\boldsymbol{y})d\Omega_{\boldsymbol{x}} = \boldsymbol{M}(\boldsymbol{x})\boldsymbol{\alpha}(\boldsymbol{z})\Big|_{\boldsymbol{z}=\boldsymbol{x}} \tag{2.15}$$

where $\boldsymbol{M}(\boldsymbol{x})$ is the moment matrix given by

$$\boldsymbol{M}(\boldsymbol{x}) = \int_{\boldsymbol{y}\in\Omega_{\mathbf{x}}} \boldsymbol{p}(\boldsymbol{y})W(\boldsymbol{z}-\boldsymbol{y},h)\Big|_{\boldsymbol{z}=\boldsymbol{x}}\boldsymbol{p}^{\boldsymbol{T}}(\boldsymbol{y})d\Omega_{\boldsymbol{x}} \tag{2.16}$$

In numerical computations the global domain Ω_T is represented by a set of nodes. The integrals in equations (2.14) and (2.15) are evaluated using those nodes inside $\Omega_{\boldsymbol{x}}$ as quadrature points. Consequently, the parameters $\boldsymbol{\alpha}$ are obtained with the following equation

$$\boldsymbol{\alpha}(\boldsymbol{z})\Big|_{\boldsymbol{z}=\boldsymbol{x}} = \boldsymbol{M}^{-\mathbf{1}}(\boldsymbol{x})\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\boldsymbol{W}(\boldsymbol{x})\boldsymbol{u}_{\boldsymbol{\Omega_x}} \tag{2.17}$$

where the matrix $\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}$ is defined as the basis functions evaluated at the nodes inside the compact support, and the vector $\boldsymbol{u}_{\Omega_{\boldsymbol{x}}}$ contains the n_x pointwise values of the function to be reproduced, $\boldsymbol{u}(\boldsymbol{x})$ inside the compact domain $\boldsymbol{u}_{\Omega_{\boldsymbol{x}}}$, schematically represented in Figure 2.6.

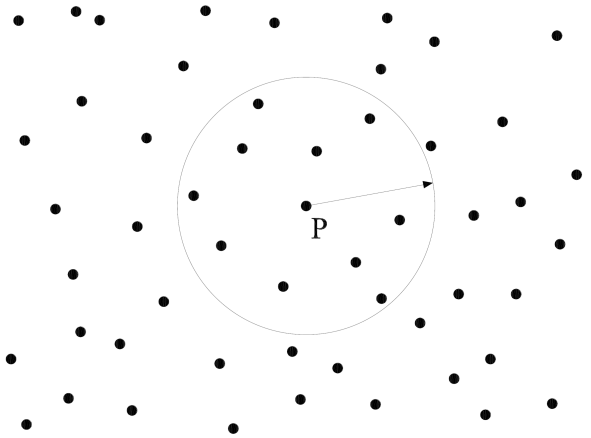


Figure 2.6: Compact support ($\Omega_{\boldsymbol{x}}$) centered in point P.

Equation (2.18) represents mathematically the previous statement.

$$\boldsymbol{u_{\Omega_x}} = (u(\boldsymbol{x_1}), u(\boldsymbol{x_2}), \ldots, u(\boldsymbol{x_{n_x}}))^T \tag{2.18}$$

The moment matrix is given by

$$\boldsymbol{M(x)} = \boldsymbol{P}_{\Omega_x}\boldsymbol{W(x)}\boldsymbol{P}_{\Omega_x}^T \tag{2.19}$$

It is important to notice that the dimension of the moment matrix does not depend of the number of points that are employed (n_x). The dimension of this matrix only depends on the number of elements of the basis function vector, m. $\boldsymbol{P_{\Omega_x}}$ is a matrix of dimension $m \times n_x$ and the dimension of $\boldsymbol{W(x)}$ is $n_x \times n_x$. Therefore, the dimension of the moment matrix is $m \times m$. Finally, the MLS interpolation structure can be identified as

$$\widehat{u}(\boldsymbol{x}) = \boldsymbol{p}^T(\boldsymbol{x})\boldsymbol{M^{-1}(x)}\boldsymbol{P}_{\Omega_x}\boldsymbol{W(x)}\boldsymbol{u}_{\Omega_x} = \sum_{k=1}^{n_{\mathbf{x}}} N_k(\boldsymbol{x})u_k \tag{2.20}$$

Computationally, in order to prevent the matrix $\boldsymbol{M}$ from being singular or ill conditioned, the cloud of neighbors should fulfill some requirements. If the number of neighbors is less than m, the moment matrix becomes singular. If the number of neighbors is $n_x = m$, the moment matrix can be ill conditioned, so it is convenient in practice to use a number of neighbors greater than m with the information coming from as many directions as possible.

In analogy to the finite elements method, the approximation can be written in terms of the MLS shape functions, $\boldsymbol{N}(\boldsymbol{x})$. These shape functions depend on the distribution of the set of nodes employed, so that if the distribution is time independent, the shape functions will not change over time.

From equation (2.20) the shape functions at $\boldsymbol{x}$, $\boldsymbol{N}(\boldsymbol{x})$, can be obtained as:

$$\boldsymbol{N}^T(\boldsymbol{x}) = \boldsymbol{p}^T(\boldsymbol{x})\boldsymbol{M^{-1}(x)}\boldsymbol{P}_{\Omega_x}\boldsymbol{W(x)} \tag{2.21}$$

To reduce the ill-conditioned state of the matrix $\boldsymbol{M}$, the basis functions are evaluated locally, as suggested in [147]. That is, if the shape function vector, $\boldsymbol{N}(\boldsymbol{x})$, is evaluated at $\boldsymbol{x_I}$, the basis function will be evaluated at $\boldsymbol{p}(\frac{\boldsymbol{x}-\boldsymbol{x_I}}{h})$, where h is the smoothing length. So equation (2.21) will be defined as

$$\boldsymbol{N}^T(\boldsymbol{x_I}) = \boldsymbol{p}^T(\boldsymbol{0})\boldsymbol{M^{-1}(x_I)}\boldsymbol{P}_{\Omega_{x_I}}\boldsymbol{W(x_I)} \tag{2.22}$$

Equations (2.23), (2.24) and (2.25) show the two-dimensional lineal, quadratic and cubic basis functions for MLS.

$$\boldsymbol{p}(\boldsymbol{x}) = (1, x, y) \tag{2.23}$$

$$\boldsymbol{p}(\boldsymbol{x}) = (1, x, y, x^2, y^2, xy) \tag{2.24}$$

$$\boldsymbol{p}(\boldsymbol{x}) = (1, x, y, x^2, y^2, xy, x^2y, xy^2, x^3, y^3) \tag{2.25}$$

In the above expressions, (x, y) denotes the cartesian coordinates of $\boldsymbol{x}$.

2.3.1 Computational derivatives

The high-order approximate derivatives of the field variable $u(\boldsymbol{x})$ evaluated at $\boldsymbol{x} = \boldsymbol{x_I}$ can be expressed in terms of the derivatives of the MLS shape function.

$$\left.\frac{\partial u}{\partial x_i}\right|_{\boldsymbol{x}=\boldsymbol{x_I}} \approx \left.\frac{\partial \widehat{u}}{\partial x_i}\right|_{\boldsymbol{x}=\boldsymbol{x_I}} = \sum_{k=1}^{n_{\mathbf{x}}} \left.\frac{\partial N_k(\boldsymbol{x})}{\partial x_i}\right|_{\boldsymbol{x}=\boldsymbol{x_I}} u_k \tag{2.26}$$

$$\left.\frac{\partial^2 u}{\partial x_i^2}\right|_{\boldsymbol{x}=\boldsymbol{x_I}} \approx \left.\frac{\partial^2 \widehat{u}}{\partial x_i^2}\right|_{\boldsymbol{x}=\boldsymbol{x_I}} = \sum_{k=1}^{n_{\mathbf{x}}} \left.\frac{\partial^2 N_k(\boldsymbol{x})}{\partial {x_i}^2}\right|_{\boldsymbol{x}=\boldsymbol{x_I}} u_k \tag{2.27}$$

So the *n-th* derivative will be:

$$\left.\frac{\partial^n u}{\partial x_i^n}\right|_{\boldsymbol{x}=\boldsymbol{x_I}} \approx \left.\frac{\partial^n \widehat{u}}{\partial {x_i}^n}\right|_{\boldsymbol{x}=\boldsymbol{x_I}} = \sum_{k=1}^{n_{\mathbf{x}}} \left.\frac{\partial^n N_k(\boldsymbol{x})}{\partial {x_i}^n}\right|_{\boldsymbol{x}=\boldsymbol{x_I}} u_k \tag{2.28}$$

As seen in equations (2.26), (2.27) and (2.28), in order to obtain the gradient and its successive derivatives there is a need to compute the derivatives of the shape function. As equation (2.22) establishes, the shape function depends on the basis of functions, the matrix $\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}$ and the *kernel*. Since the matrix $\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}$ is computed from the basis evaluated at the given points, it does not depend on the variable $\boldsymbol{x}$. So the gradient will be given by

$$\begin{aligned}\left.\frac{\partial N^T(\boldsymbol{x})}{\partial x_i}\right|_{\boldsymbol{x}=\boldsymbol{x_I}} &= \frac{\partial \boldsymbol{p}^T(\mathbf{0})}{\partial x_i}\boldsymbol{M^{-1}}(\boldsymbol{x_I})\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\boldsymbol{W}(\boldsymbol{x_I}) + \\ +\boldsymbol{p}^T(\mathbf{0})\frac{\partial \boldsymbol{M}^{-1}(\boldsymbol{x_I})}{\partial x_i}\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\boldsymbol{W}(\boldsymbol{x_I}) &+ \boldsymbol{p}^T(\mathbf{0})\boldsymbol{M}^{-1}(\boldsymbol{x_I})\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\frac{\partial \boldsymbol{W}(\boldsymbol{x_I})}{\partial x_i}\end{aligned} \tag{2.29}$$

Where the matrix $\boldsymbol{M}(\boldsymbol{x})$ was defined in (2.19), so we can rewrite equation (2.30) as

$$\left.\frac{\partial N^T(\boldsymbol{x})}{\partial x_i}\right|_{\boldsymbol{x}=\boldsymbol{x_I}} = \frac{\partial \boldsymbol{p}^T(\boldsymbol{0})}{\partial x_i}\boldsymbol{M}^{-1}(\boldsymbol{x_I})\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\boldsymbol{W}(\boldsymbol{x_I}) + \boldsymbol{p}^T(\boldsymbol{0})\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}^{-T}\frac{\partial \boldsymbol{W}^{-1}(\boldsymbol{x_I})}{\partial x_i}\boldsymbol{W}(\boldsymbol{x_I}) + \tag{2.30}$$
$$+\boldsymbol{p}^T(\boldsymbol{0})\boldsymbol{M}^{-1}(\boldsymbol{x_I})\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\frac{\partial \boldsymbol{W}(\boldsymbol{x_I})}{\partial x_i}$$

And the second derivate,

$$\begin{aligned}\left.\frac{\partial^2 N^T(\boldsymbol{x})}{\partial x_i^2}\right|_{\boldsymbol{x}=\boldsymbol{x_I}} &= \frac{\partial^2 \boldsymbol{p}^T(\boldsymbol{0})}{\partial x_i^2}\boldsymbol{M}^{-1}(\boldsymbol{x_I})\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\boldsymbol{W}(\boldsymbol{x_I}) + \frac{\partial \boldsymbol{p}^T(\boldsymbol{0})}{\partial x_i}\frac{\partial \boldsymbol{M}^{-1}(\boldsymbol{x_I})}{\partial x_i}\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\boldsymbol{W}(\boldsymbol{x_I}) + \\ &+\frac{\partial \boldsymbol{p}^T(\boldsymbol{0})}{\partial x_i}\boldsymbol{M}^{-1}(\boldsymbol{x_I})\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\frac{\partial \boldsymbol{W}(\boldsymbol{x_I})}{\partial x_i} + \boldsymbol{p}^T(\boldsymbol{0})\frac{\partial^2 \boldsymbol{M}^{-1}(\boldsymbol{x_I})}{\partial {x_i}^2}\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\boldsymbol{W}(\boldsymbol{x_I}) + \\ &+\frac{\partial \boldsymbol{p}^T(\boldsymbol{0})}{\partial x_i}\frac{\partial \boldsymbol{M}^{-1}(\boldsymbol{x_I})}{\partial x_i}\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\boldsymbol{W}(\boldsymbol{x_I}) + \boldsymbol{p}^T(\boldsymbol{0})\frac{\partial \boldsymbol{M}^{-1}(\boldsymbol{x_I})}{\partial x_i}\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\frac{\partial \boldsymbol{W}(\boldsymbol{x_I})}{\partial x_i} + \\ &+\frac{\partial \boldsymbol{p}^T(\boldsymbol{0})}{\partial x_i}\boldsymbol{M}^{-1}(\boldsymbol{x_I})\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\frac{\partial \boldsymbol{W}(\boldsymbol{x_I})}{\partial x_i} + \boldsymbol{p}^T(\boldsymbol{0})\frac{\partial \boldsymbol{M}^{-1}(\boldsymbol{x_I})}{\partial x_i}\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\frac{\partial \boldsymbol{W}(\boldsymbol{x_I})}{\partial x_i} + \\ &+\boldsymbol{p}^T(\boldsymbol{0})\boldsymbol{M}^{-1}(\boldsymbol{x_I})\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\frac{\partial^2 \boldsymbol{W}(\boldsymbol{x_I})}{\partial {x_i}^2}\end{aligned} \tag{2.31}$$

The computation of the second and successive derivatives leads to large coding effort and computational cost. As suggested in [147], the second and successive derivatives of the shape functions can be approximated by the diffuse ones. In the diffuse approach, the derivatives of the term $\boldsymbol{M}^{-1}(\boldsymbol{x})\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\boldsymbol{W}(\boldsymbol{x})$ are neglected. Recently, a semi-diffuse approach was presented in [22]. The basic idea is the use of the derivatives already computed in equation (2.30) to obtain the value of $\frac{\partial^2 N^T(\boldsymbol{x})}{\partial x_i^2}$. In this thesis, the diffuse approach is employed, so the second derivatives are computed as

$$\left.\frac{\partial^2 N^T(\boldsymbol{x})}{\partial x_i^2}\right|_{\boldsymbol{x}=\boldsymbol{x_I}} \approx \frac{\partial^2 \boldsymbol{p}^T(\boldsymbol{0})}{\partial x_i^2}\boldsymbol{M^{-1}}(\boldsymbol{x_I})\boldsymbol{P}_{\Omega_{\boldsymbol{x}}}\boldsymbol{W}(\boldsymbol{x_I}) \tag{2.32}$$

It has been shown that the diffuse approach converges at an optimal rate to the exact derivatives [39, 147], but it looses some accuracy. However the computational cost and the complexity decrease.

2.3.2 The smoothing function or *kernel*

The *kernel* function has a dual mission. The first one is to weigh the different points that take place in the interpolation according to the distance to the point of interest $\boldsymbol{x_I}$. The second role is to ensure that nodes leave or enter the support domain in a smooth

manner when the point of interest $\boldsymbol{x_I}$ moves. A wide variety of kernel function are described in the literature [121, 122, 123, 140, 169], most of them are splines or exponential functions. In [169], a deep description of the exponential, cubic and regularized cubic kernels with the influence of the different parameters is presented. For example, the exponential kernel for the one-dimensional case is defined as

$$W_k(x_k, x_I, s_x) = \frac{e^{-\left(\frac{d}{c}\right)^2} - e^{-\left(\frac{d_m}{c}\right)^2}}{1 - e^{-\left(\frac{d_m}{c}\right)^2}} \tag{2.33}$$

for $k = 1, ..., n_x$, where $d = |x_k - x_I|$, $d_m = 2\max(|x_k - x_I|)$, d_m is the *smoothing length*, n_x the number of neighbors and x_I is the reference point where the compact support is centered.

In equation (2.33), the coefficient c is introduced, which is defined as $c = \frac{d_m}{s_x}$, where s_x is the shape parameter of the *kernel*. This parameter plays an important role, since it defines the properties of the *kernel* and therefore, the properties of the numerical scheme [147]. More details can be found in [147, 149, 169].

In Figure 2.7, the *kernel* and its first derivate are represented for different values of the shape parameter s_x.

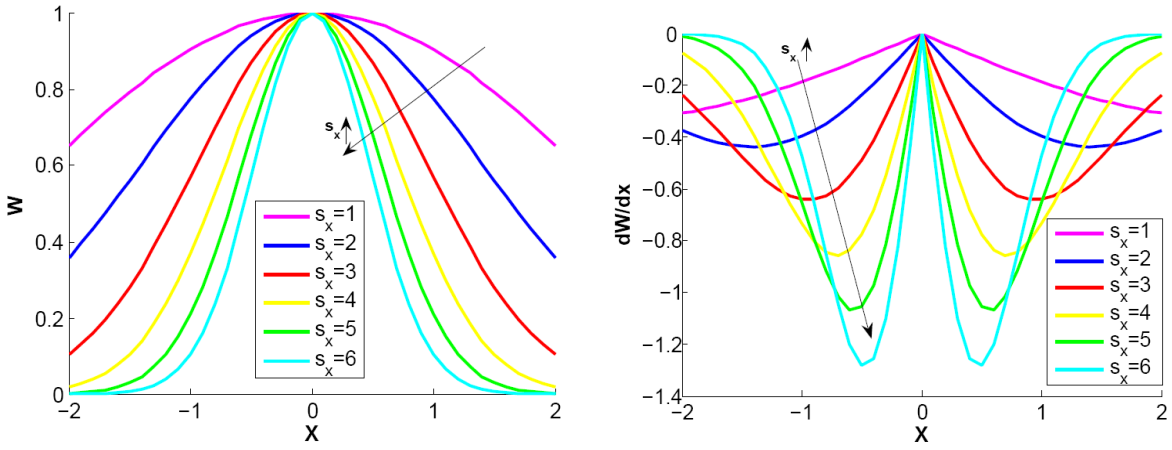


Figure 2.7: Influence of the parameter s_x in the exponential *kernel* and its first derivative

A n-dimensional *kernel* can be obtained by multiplying n 1D *kernels*. Thus, the 2D exponential *kernel* is defined as

$$W_k(\boldsymbol{x}_k, \boldsymbol{x}_I, s_x, s_y) = W_k(x_k, x_I, s_x) W_k(y_k, y_I, s_y) \tag{2.34}$$

2.4 The FV-MLS method

The FV-MLS method was introduced in [36, 37, 39, 40, 147, 153, 169]. The basic idea of this finite volume method is the use of Moving Least Squares approximants to obtain the high order reconstruction of the variables, equation (2.11), needed for the evaluation of the fluxes of equation (2.8).

On the previous section the MLS approximants were obtained using a set of nodes where the value of the variables was known (schematically represented in Figure 2.6). In a cell-centered finite volume approach, the variables are stored at cell-centroid, as already commented in section 1. Thus, in the FV-MLS method, the integral defined in (2.14) is evaluated at cell centroids. The compact support is formed with a set of n_x cells that will be denoted from now on as stencil. An example of a typical FV-MLS stencil is plotted in Figure 2.8, where the shaded cells represents the stencil for cell I. Note that the cell I also belongs to the stencil.

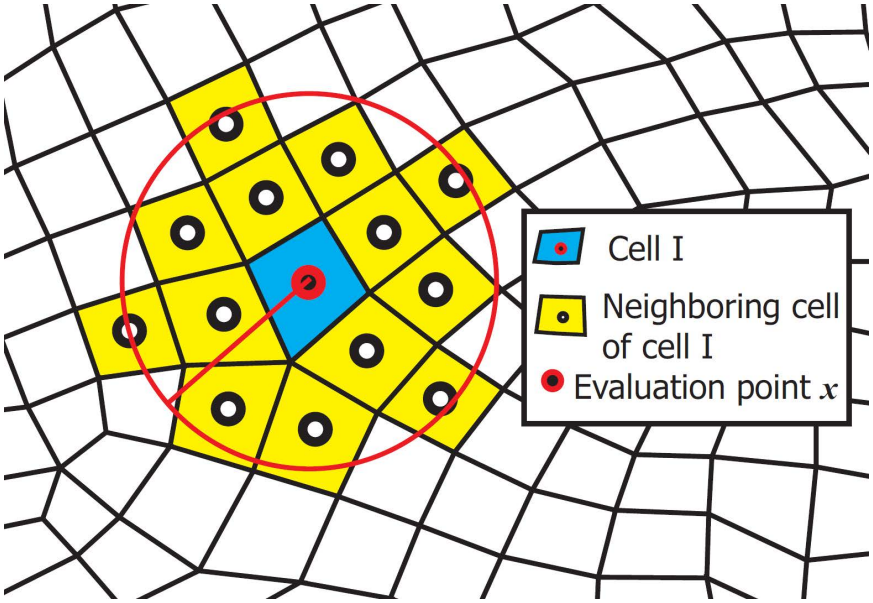


Figure 2.8: Typical stencil for interior cells used for cubic MLS approximation centered at cell centroid I. Shaded cells represent the stencil of cell I

For the sake of clarity, $\boldsymbol{N}$ refers to the shape functions computed at a cell centroid, using stencils such as the one shown in Figure 2.9, and $\boldsymbol{N^g}$ for the shape functions computed at an integration point, using stencils such as the shown in Figure 2.10. Note that the only difference between $\boldsymbol{N}$ and $\boldsymbol{N^g}$ is the location where the shape functions are evaluated, but they are computed using the same computational procedure.

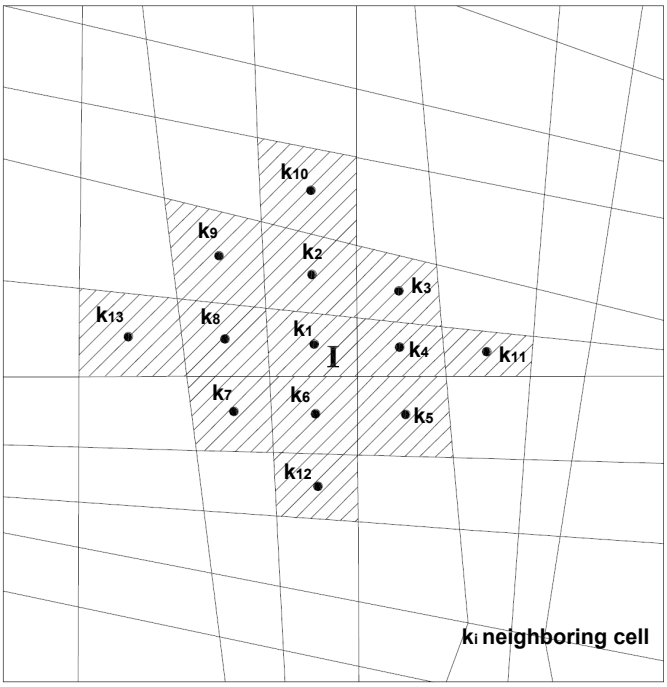


Figure 2.9: Typical stencil for interior cells used for cubic MLS approximation centered at cell centroid I. k_i represents the neighboring cells of cell I

The FV-MLS method acknowledges the different nature of the elliptic and hyperbolic terms. The convective terms are computed at the integration points using a broken reconstruction, mathematically described as $\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}} = \boldsymbol{\mathcal{H}}(u^+, u^-)$. A schematic representation of the reconstructed variables of the inviscid fluxes is shown in Figure 2.11.

The elliptic terms are computed directly at integration points with MLS. Note that the MLS approximation is centered at the integration points, giving a high-order continuous reconstruction. The accuracy of the interpolation is clearly improved when MLS approximants are centered at the integration point on a unstructured mesh [147].

The cell-wise broken reconstruction defined for the hyperbolic fluxes is actually a piecewise continuous approximation to $u(\boldsymbol{x})$. The reconstruction is obtained using a Taylor series expansion, that for a steady state case reads as

$$u(\boldsymbol{x})\Big|_{\boldsymbol{x}=\boldsymbol{x}_j} = u_I + \boldsymbol{\nabla} u_I \cdot (\boldsymbol{x}_j - \boldsymbol{x}_I) + \frac{1}{2} (\boldsymbol{x}_j - \boldsymbol{x}_I)^T \boldsymbol{H} \ (\boldsymbol{x}_j - \boldsymbol{x}_I) \tag{2.35}$$

As seen in equation (2.35), the Taylor series expansion, needed for the computation of the hyperbolic flux $\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}}(u_I, \boldsymbol{\nabla} u_I, ...)$, depends of the first and successive derivatives computed at cell centroid. For example, the gradient at cell centroid is given by

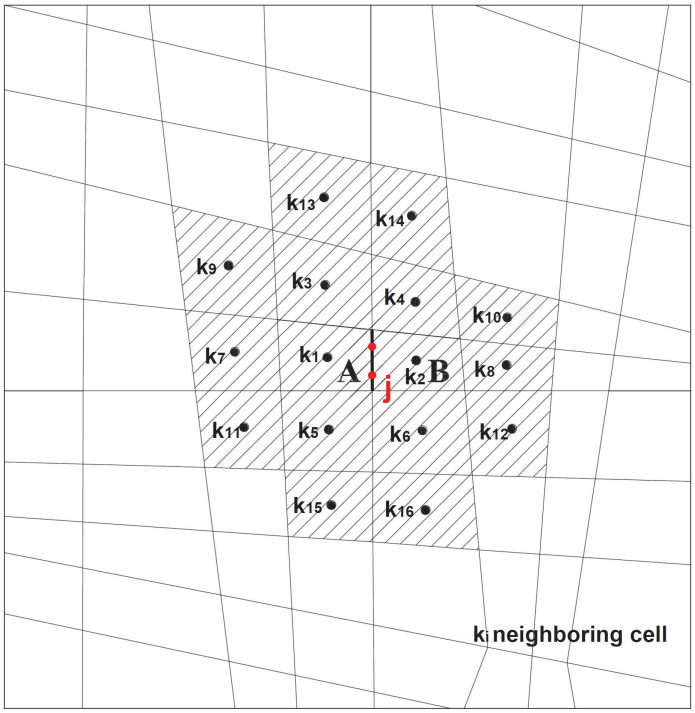


Figure 2.10: Stencil for interior cells used for cubic MLS approximation centered at integration point j. k_i represents the neighboring cells of the integration point j

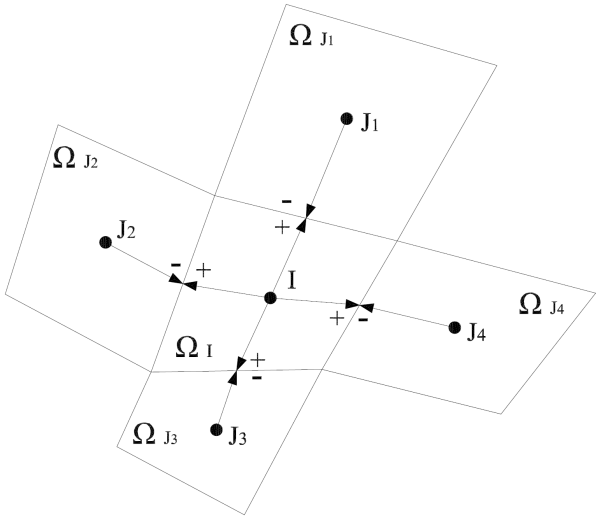


Figure 2.11: Reconstructed variables used to evaluate the inviscid fluxes at the integration point at the edges of the control volume Ω_I. From [37].

$$\nabla u_I = \sum_{k=1}^{n_{\boldsymbol{x}}} \nabla N_k(\boldsymbol{x}_I) u_k \tag{2.36}$$

For the elliptic fluxes $\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{V}}}(u_j, \boldsymbol{\nabla} u_j, ...)$, the values of u_j and $\boldsymbol{\nabla} u_j$ are computed as

$$u_j = \sum_{k=1}^{n_{\boldsymbol{x}}} N_k^g(\boldsymbol{x}_j) u_k \tag{2.37}$$

$$\nabla u_j = \sum_{k=1}^{n_{\boldsymbol{x}}} \nabla N_k^g(\boldsymbol{x}_j) u_k \tag{2.38}$$

The dual continuous/discontinuous reconstruction of the solution is crucial in order to obtain accurate and efficient numerical schemes for mixed parabolic/hyperbolic problems. The advantage is that it allows to make use of Riemann solvers, limiters, and other standard finite volume technologies.

The definition of the stencil is a crucial point. The stencil should be as compact as possible, and the selection of neighbors must be suitable for general grids. It is important to remark that these stencils are typically centered around the cell, as the stencil in Figure 2.9. This fact avoids the spatial bias which is often found in patch-based piece-wise polynomial approximations.

The non-interpolant nature of MLS is important for the imposition of boundary conditions. For boundary cells, the stencil comprises the surrounding cells including the so-called *ghost cells* [39], as the stencil defined in Figure 2.12. The *ghost cells* are included at midpoint of the boundary edge in order to improve the computation of MLS approximations at the boundaries. During the simulation, the variables at these locations will be either extrapolated or assigned a certain value, depending on the type of boundary condition to be enforced.

2.5 Conclusions

In this Chapter the basic ingredients of the thesis are introduced. First, the basic Finite Volume formulation for a generic conservation law is studied. Then, Moving Least Squares is presented. On a finite volume framework, the main problem for achieving high-order accuracy on unstructured meshes is the computation of the gradients and successive

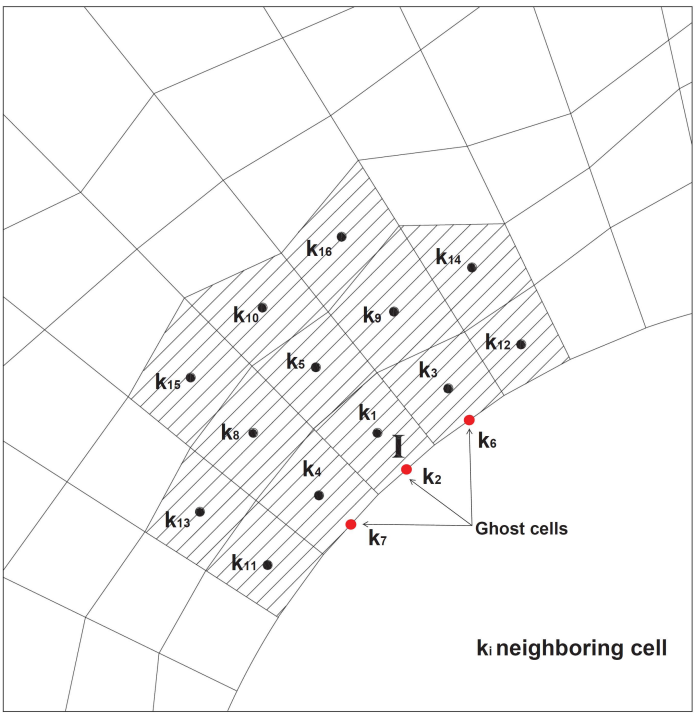


Figure 2.12: MLS stencil for boundary cells. Note that ghost cells are included in the stencil of cell I.

derivatives needed for the reconstruction of the variables. In order to circumvent this problem the MLS approximation is employed to develop a high-order finite volume method called FV-MLS.

Once the basis of the FV-MLS is explained, the FV-MLS will be used for the resolution of the compressible Navier-Stokes for all speed flows applying a density-based algorithm in Chapter 3. Next, in Chapter 4, the incompressibility assumption will be considered, and therefore the incompressible Navier-Stokes will be analyzed. Finally, in Chapter 5, three high-order sliding mesh techniques based on the use of Moving Least Squares are presented and analyzed.

Chapter 3

A high-order FV-MLS formulation for compressible flows at all-speed.

3.1 Introduction

Traditionally, two families of finite volume schemes have been developed to compute either compressible or incompressible flows. Density-based solvers [34, 181, 213] are used for the computation of flows when compressibility effects are important (mainly transonic, supersonic and hypersonic flows), whereas pressure-based solvers [29, 159, 164] are designed to compute incompressible flows. In both techniques, the velocity is obtained from the momentum equations. The difference between the two approaches is the computation of the pressure field. In density-based solvers, the density is computed from the continuity equation and then the pressure is obtained from an equation of state. Pressure-based solvers solve a Poisson-type equation, obtained from continuity and momentum equations, to compute the pressure.

In pressure-based solvers, the SIMPLE and related algorithms [159] have been widely used to compute incompressible flows, and several authors have developed methods to extend these algorithms to all-speed flows [101, 141].

From a practical point of view, density-based schemes are not suitable for flows with Mach number lower than 0.3 [182]. These solvers present a number of problems: stiffness of the equations, cancellation in the pressure variable [187] and the loss of accuracy due to a wrong numerical diffusion (accuracy problem).

However, it is preferable the development of solvers useful for all the regimes of a flow, not only for user's convenience, but also because the importance of flows where low and high Mach regions are present, such as the flow inside a gas and a steam turbine, flow

past an aerodynamic profile at high angle of attack, or flow past a blunt body, or when compressibility effects are important, even in low Mach number flows, for example for the simulation of cavitating flows on a pump or a hydraulic turbine. Thus, the modification of density or pressure-based solvers to compute all-speed flows is a current active area of research.

It is known that the accuracy problem of density-based solvers in low-Mach flows is originated by the introduction of spurious pressure and velocity waves that avoid the velocity field to verify (or at least be close to) the zero-divergence constraint [70, 69, 70, 43, 105, 116]. Preconditioning techniques [28, 69, 70, 208, 215] have been developed to increase the accuracy (and also to overcome the stiffness problem) of Godunov schemes in low Mach flows. The preconditioning matrix multiplies the time derivatives of the set of equations, with the effect of re-scaling the eigenvalues (acoustic-speed) of the system, but paying the price of spoiling the temporal accuracy of the scheme. Thus, this technique was initially developed for steady flows, but extensions to unsteady flow have been proposed by using a dual time stepping technique [132, 219]. An Asymptotic-Preserving methodology has been presented in [35].

Recently, it was reported [177] that the accuracy problem is not observed when the Roe's approximate Riemann solver is used in a first-order finite volume scheme on triangular grids. This behavior is related to cancellation of the normal velocity jump. Based on this finding, a fix for the numerical flux of Roe is presented in [179]. This fix reduces the jumps in the normal velocity component of the Roe's Riemann solver. Other fixes have also been proposed for Roe flux [203] and for other Riemann solvers, such as the HLL-family [117]. Several flux-splitting type schemes accurate at low and also high Mach have been presented in [120, 190, 191]. On the other hand, some authors have reported Discontinuous Galerkin (DG) solutions of low-Mach flow with and without preconditioning [7, 55, 127]. The high-order approach reduces the jumps in the normal velocity component alleviating the accuracy problem. It has been shown, however, that in DG schemes the accuracy problem is not completely solved [7], since a grid dependency with the Mach number still remains.

In this Chapter, a high-order density-based finite-volume formulation for all-speed flows is developed, based on the FV-MLS method presented in Chapter 2. In order to solve the accuracy problem, two approaches has been addressed: the increase of the order of the numerical scheme and the use of fixes to the Roe and Rusanov numerical fluxes.

The outline of this Chapter is as follows. In section 3.2 the general formulation of the proposed all-speed scheme is presented. Then, in section 3.3 the accuracy of the numerical scheme is studied. In section 3.6 the computations for an unsteady transonic

viscous flow over a circular cylinder are presented.

3.2 General formulation

Following the conservation law presented in equation (2.1) the Navier-Stokes equations can be rewritten as

$$\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{\nabla} \cdot \left(\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}} - \boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{V}}} \right) = \boldsymbol{S} \qquad in \quad \Omega_T \tag{3.1}$$

where $\boldsymbol{U}$ is the vector of variables, $\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}}$ is the inviscid flux vector, $\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{V}}}$ is the viscous flux vector and $\boldsymbol{S}$ is a source term. If the viscous flux, $\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{V}}}$, is neglected, the Euler equations are obtained. The set of equations needs to be supplemented with suitable initial and boundary conditions. These are a crucial point in low-Mach computations using the compressible system of equations. It has been shown [43, 69, 186] that a set of "well-prepared" initial conditions is required for the convergence of the solution of the compressible system of equations to the solution of the incompressible set when $M \to 0$. In this context, "well-prepared" means that the initial pressure field scales with the square of the Mach number and that the initial velocity field is close to a divergence free field [69]. In addition, Dirichlet boundary conditions may also lead to inaccurate results of low Mach finite volume schemes [43].

Applying the finite volume discretization presented in Chapter 2, the final discrete scheme for the continuous/discontinuous approach can be written as

$$\Omega_I \frac{\partial \boldsymbol{U}_I}{\partial t} + \sum_{j=1}^{N_f} \sum_{ig=1}^{N_G} \left[\boldsymbol{\mathcal{H}}(\boldsymbol{U}_j^+, \boldsymbol{U}_j^-, \hat{\boldsymbol{n}}_j) - \boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{V}}}{}_j \cdot \hat{\boldsymbol{n}}_j \right]_{ig} \mathcal{W}_{ig} = \int_{\Omega_I} S \, d\Omega \tag{3.2}$$

where $\boldsymbol{\mathcal{H}}(\boldsymbol{U}_j^+, \boldsymbol{U}_j^-, \hat{\boldsymbol{n}}_j)$ is a suitable numerical flux. The numerical fluxes will be described in section 3.2.2.

3.2.1 Time integration

In this Chapter a third-order Runge-Kutta scheme of Shu and Osher [192] is used in the time discretization. The scheme can be written in the following standard form

$$\begin{aligned} \boldsymbol{U}_I^{(1)} &= \boldsymbol{U}_I^{(n)} + \Delta t \mathcal{L}(\boldsymbol{U}_I^{(n)}) \\ \boldsymbol{U}_I^{(2)} &= \frac{3}{4}\boldsymbol{U}_I^{(n)} + \frac{1}{4}\boldsymbol{U}_I^{(1)} + \frac{1}{4}\Delta t \mathcal{L}(\boldsymbol{U}_I^{(1)}) \\ \boldsymbol{U}_I^{(n+1)} &= \frac{1}{3}\boldsymbol{U}_I^{(n)} + \frac{2}{3}\boldsymbol{U}_I^{(2)} + \frac{2}{3}\Delta t \mathcal{L}(\boldsymbol{U}_I^{(2)}) \end{aligned} \tag{3.3}$$

where the superscripts n and $n+1$ denote the time iteration, 1, the superscripts 1 and 2 denotes the Runge-Kutta inner steps and Δt is the time step. The operator $\mathcal{L}(\boldsymbol{U}_I)$ is defined as

$$\mathcal{L}(\boldsymbol{U}_I) = \int_{\Omega_I} S \, d\Omega - \sum_{j=1}^{N_f} \sum_{ig=1}^{N_G} \left[\boldsymbol{\mathcal{H}}(\boldsymbol{U}_j^+, \boldsymbol{U}_j^-, \hat{\boldsymbol{n}}_j) - \boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{V}}}{}_j \cdot \hat{\boldsymbol{n}}_j \right]_{ig} \mathcal{W}_{ig} \tag{3.4}$$

3.2.2 Numerical Fluxes

As pointed in section 2.4, the FV-MLS method acknowledges the different nature of the elliptic and hyperbolic terms. The dual continuous/discontinuous reconstruction of the solution is crucial in order to obtain accurate and efficient numerical schemes for mixed parabolic/hyperbolic problems.

The FV-MLS computes the elliptic fluxes using a continuous reconstruction, that is, the fluxes are computed directly at integration points applying the MLS reconstruction.

The convective terms are computed at integration points using a broken reconstruction, mathematically described as $\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}} = \boldsymbol{\mathcal{H}}(\boldsymbol{U}_j^+, \boldsymbol{U}_j^-, \hat{\boldsymbol{n}}_j)$. As mentioned in Chapter 2, the use of a broken reconstruction does not have a unique solution, since there is a discontinuity at cell faces $\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}}(\boldsymbol{U}_j^+) \neq \boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}}(\boldsymbol{U}_j^-)$. Godunov-type schemes for hyperbolic problems rely on the solution of a Riemann problem with the two states.

3.2.3 Approximate Riemann solvers

Even tough computing the exact solution of the Riemann problem is the best methodology, it can be complicated and numerically expensive. In order to circumvent this problem, several authors have replace the solution of the exact Riemann problem by an approximation. In this Chapter, two approximated Riemann solvers are studied.

- The Roe flux: The Roe flux belongs to the category of Flux Difference Splitting [42]. The basic idea of the Flux Difference Splitting is to linearize the hyperbolic flux $\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}}$ around the cell interface. The original method was proposed by Roe in [181].

Nowadays, it is one of the most popular numerical flux. The Roe flux [181] can be written as

$$\boldsymbol{\mathcal{H}}_j = \frac{1}{2}(\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}}(\boldsymbol{U}_j^+) + \boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}}(\boldsymbol{U}_j^-)) \cdot \hat{\boldsymbol{n}} - \frac{1}{2}\sum_{k=1}^{4} \tilde{\alpha}_k |\tilde{\lambda}_k| \tilde{\boldsymbol{r}}_k \tag{3.5}$$

In equation (3.5) $\tilde{\lambda}_k$ and $\tilde{\boldsymbol{r}}_k$ are the eigenvalues and eigenvectors of the approximated Jacobian [181] defined as

$$\begin{aligned} \tilde{\lambda}_1 &= \tilde{\boldsymbol{v}} \cdot \boldsymbol{n} - \tilde{c} \\ \tilde{\lambda}_2 &= \tilde{\lambda}_3 = \tilde{\boldsymbol{v}} \cdot \boldsymbol{n} \\ \tilde{\lambda}_4 &= \tilde{\boldsymbol{v}} \cdot \boldsymbol{n} + \tilde{c} \end{aligned} \tag{3.6}$$

$$[\tilde{\boldsymbol{r}}_1, \tilde{\boldsymbol{r}}_2, \tilde{\boldsymbol{r}}_3, \tilde{\boldsymbol{r}}_4] = \begin{pmatrix} 1 & 0 & 1 & 0 \\ \tilde{u} - \tilde{c}n_x & -\tilde{c}n_y & \tilde{u} & \tilde{u} + \tilde{c}n_x \\ \tilde{v} - \tilde{c}n_y & -\tilde{c}n_x & \tilde{v} & \tilde{v} + \tilde{c}n_y \\ \tilde{H} - \tilde{c}\,\tilde{\boldsymbol{v}} \cdot \boldsymbol{n} & \tilde{c}(\tilde{v}n_x - \tilde{u}n_y) & \frac{1}{2}(\tilde{u}^2 + \tilde{v}^2) & \tilde{H} + \tilde{c}\,\tilde{\boldsymbol{v}} \cdot \boldsymbol{n} \end{pmatrix} \tag{3.7}$$

where H is the enthalpy, c is the sound velocity and $\tilde{\boldsymbol{v}} = (\tilde{u}, \tilde{v})^T$ is the velocity vector. We also define $\tilde{\alpha}_k$ as

$$\begin{aligned} \tilde{\alpha}_1 &= \tfrac{1}{2\tilde{c}^2}\left[\Delta(p) - \tilde{\rho}\tilde{c}\,(\Delta(u)n_x + \Delta(v)n_y)\right] \\ \tilde{\alpha}_2 &= \tfrac{\tilde{\rho}}{\tilde{c}}\left[\Delta(v)n_x - \Delta(u)n_y\right] \\ \tilde{\alpha}_3 &= \tfrac{1}{\tilde{c}^2}\left[\Delta(p) - \tilde{c}^2\Delta(\rho)\right] \\ \tilde{\alpha}_4 &= \tfrac{1}{2\tilde{c}^2}\left[\Delta(p) + \tilde{\rho}\tilde{c}\,(\Delta(u)n_x + \Delta(v)n_y)\right] \end{aligned} \tag{3.8}$$

Symbol $\tilde{\cdot}$ indicates Roe's average [181], and $\Delta(\cdot) = (\cdot)^- + (\cdot)^+$.

- The Rusanov flux: The Rusanov flux is a simple example of the HLL Riemann solvers. This class of robust approximate Riemann solvers are based on the construction of averaging intermediate states of the Riemann problem. The Rusanov flux is very efficient, robust and easy to implement. However the Rusanov flux is more diffusive than the Roe flux and it is considered an incomplete Riemann solver, since does not contains the full set of waves present in the exact solution of the Riemann problem. The Rusanov flux [183] can be written as

$$\boldsymbol{\mathcal{H}}_j = \frac{1}{2}(\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}\boldsymbol{+}} + \boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}\boldsymbol{-}}) \cdot \hat{\boldsymbol{n}} - \frac{1}{2}S^+\Delta(\boldsymbol{U}) \tag{3.9}$$

with

$$S^+ = max(|\boldsymbol{v}^+| + c^+, |\boldsymbol{v}^-| + c^-) \tag{3.10}$$

In equation (3.10) $|\boldsymbol{v}|$ is the modulus of the velocity vector and $\Delta(\boldsymbol{U}) = (\boldsymbol{U}^+ - \boldsymbol{U}^-)$, represented in Figure 3.1 for a second-order scheme. Taking the viscosity parameter, S^+, as the largest local wave speed guarantees the stability of the scheme [15]. The main disadvantages is that the Rusanov scheme assumes only one intermediate wave state between two acoustic waves and S^+ corresponds to the acoustic wave speed estimate [15].

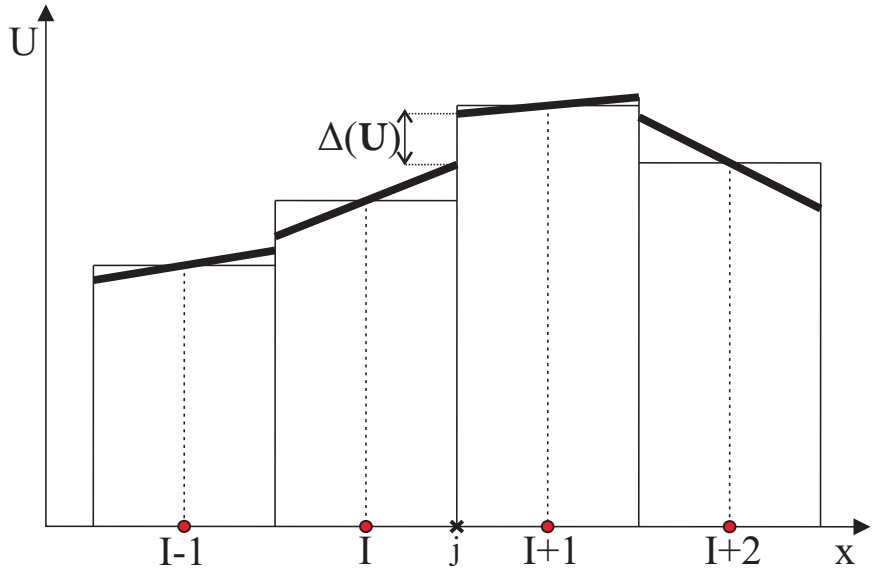


Figure 3.1: Piecewise linear reconstruction. Notice that the increase of the order of the reconstruction reduces the term $\Delta(\boldsymbol{U})$.

3.2.4 Low Mach entropy fixes

It is known that the accuracy problem of density-based solvers in low-Mach flows is originated by the introduction of spurious pressure and velocity waves that avoid the velocity field to verify (or at least be close to) the zero-divergence constraint [43, 69, 70, 105, 116]. Recently, it was reported [177] that the accuracy problem is not observed when the Roe's approximate Riemann solver is used in a first-order finite volume scheme on triangular grids. This behavior is related to cancellation of the normal velocity jump. Based on this finding, a fix for the numerical flux of Roe is presented in [179]. This fix reduces the jumps in the normal velocity component of the Roe's Riemann solver. Other fixes have also been proposed for Roe flux [203] and for other Riemann solvers, such as the HLL-family [117]. Several flux-splitting type schemes accurate at low and also high Mach have been presented in [120, 190, 191].

In this Thesis, the low Mach entropy fixes studied are: Rieper's Fix for the Roe flux [179] and Li and Gu's fix for the Rusanov flux [117].

Rieper's Fix for the Roe flux

In order to solve the accuracy problem, Rieper [179] proposes a generalization of the *low-Mach X* schemes proposed by Dellacherie [43]. It consists on a reduction of the normal velocity jump that is the term responsible of the accuracy problem. It is simply obtained by modifying the terms $\tilde{\alpha}_1$ and $\tilde{\alpha}_4$ in equation (3.8) as follows

$$\begin{array}{l} \tilde{\alpha}_1 = \frac{1}{2\tilde{c}^2}\left[\Delta(p) - \tilde{\rho}\tilde{c}f(M_l)\left(\Delta(u)n_x + \Delta(v)n_y\right)\right] \\ \tilde{\alpha}_4 = \frac{1}{2\tilde{c}^2}\left[\Delta(p) + \tilde{\rho}\tilde{c}f(M_l)\left(\Delta(u)n_x + \Delta(v)n_y\right)\right] \end{array} \tag{3.11}$$

where $f(M_l)$ is a function of the local Mach number that is active when $M_l < 1$. It is defined for a cell I as

$$f(M_l) = min(M_l, 1) \tag{3.12}$$

with

$$M_l = \frac{|\tilde{u}|_I + |\tilde{v}|_I}{\tilde{c}_I} \tag{3.13}$$

Li and Gu's fix for the Rusanov flux

Li and Gu [117] described the mechanism for what Rieper's fix and other fixes proposed work and extended the applicability of low Mach Roe fixes to HLL schemes. In this Thesis, the fix proposed in [117] is applied to the Rusanov flux. It is simply obtained by multiplying the momentum difference term in the momentum equations by the function $f(M_l)$, as

$$\Delta(\boldsymbol{U}) = \left\{ \begin{array}{l} \Delta(\rho) \\ f(M_l)\Delta(\rho v_x) \\ f(M_l)\Delta(\rho v_y) \\ \Delta(\rho E) \end{array} \right\}. \tag{3.14}$$

3.3 Obtaining physical solution using higher-order MLS reconstructions

In this section, it is shown that increasing the order of a finite volume scheme the accuracy problem is alleviated, but not completely solved, as it happens in Discontinous Galerkin schemes [7]. In order to show this fact, the Euler equations are used to compute

the potential flow past a circular cylinder at Mach numbers ranging from $M_\infty = 10^{-6}$ to $M_\infty = 10^{-1}$. High-order FV-MLS reconstructions are achieved up to 4^{th} order.

3.3.1 Accuracy assessment and pressure scaling for low-Mach computations

First, we focus on the accuracy properties of the high-order reconstructions by performing a grid refinement study using a sequence of four refined O-type meshes with regular quad cells. The coarsest grid, shown in Figure 3.2, is built from 32 points equally distributed in the circumferential direction and 16 points in the radial direction. Three additional grids (48×24, 64×32 and 96×48) are obtained by refining the coarsest mesh in both directions. The far-field is situated at 40 diameters away from the cylinder.

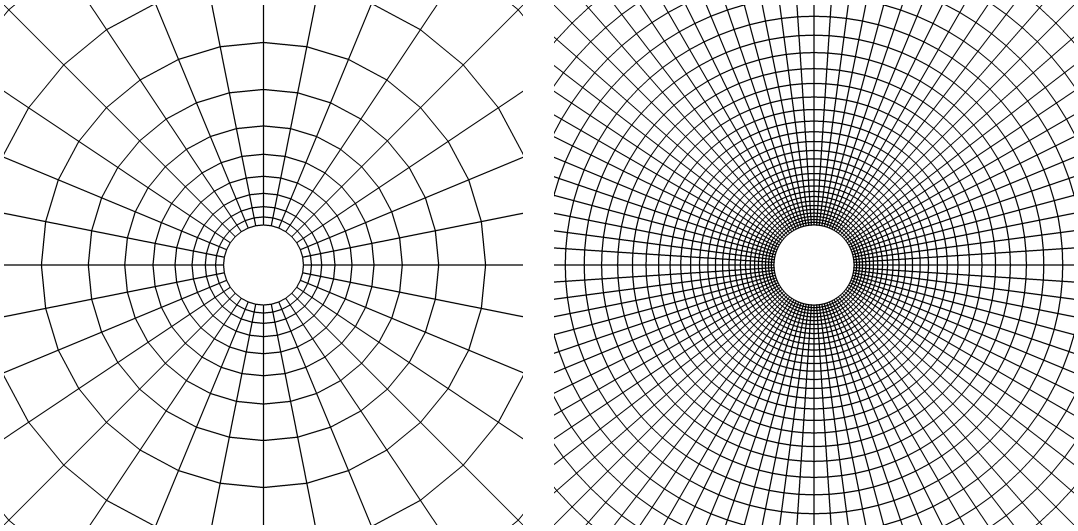


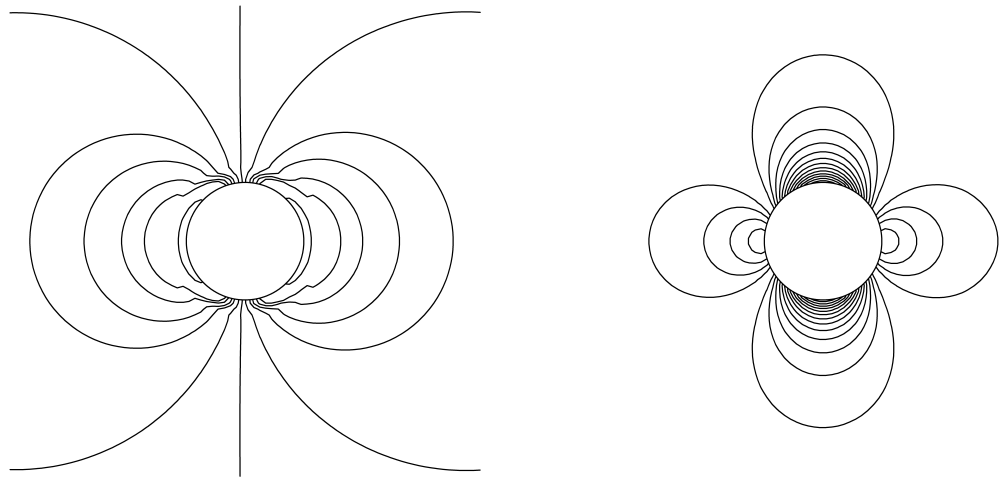
Figure 3.2: Close view of the coarsest and the finest structured O-grids employed for the computation of the inviscid flow past a circular cylinder test case. The coarsest mesh (left) has 32×16 elements and the finest (right) 96×48 elements.

All computations are initialized using a uniform flow, and they are converged until the L_2 norm of the residuals falls below 10^{-10}.

Roe scheme

Here, we investigate the effect of using higher-order MLS reconstruction schemes on the accuracy of low-Mach flows computed using the numerical flux of Roe (named hereafter the ROE-FV-MLS scheme). The freestream mach number is $M_\infty = 10^{-3}$ and the 96×48 mesh is employed. Figure 3.3 presents a comparison of the pressure contours between the

1^{st} order FV scheme and those obtained with the 4^{th} order FV-MLS method. Although the low order solution exhibits the known "*creep*" unphysical solution [117], it is clearly observed that the use of high order reconstruction permits to recover the physical solution.



a) b)

Figure 3.3: Pressure contours for inviscid flow past a cylinder test case for $M_\infty = 10^{-3}$. The solution is obtained in the 96×48 grids by using a first order Roe scheme (a) and by using the 4^{th} order FV-MLS Roe scheme (b).

Next, we study the order of accuracy of the 4^{th}-order ROE-FV-MLS scheme by computing the error in the drag coefficient for M_∞ ranging from 10^{-1} to 10^{-3}. Table 3.1 shows that the formal order of accurate is recovered for both $M_\infty = 10^{-1}$ and $M_\infty = 10^{-2}$. We however notice that obtaining physical solution for $M_\infty = 10^{-3}$ requires the use of finer grid than for lower Mach numbers. This shows that the increase in the order of a finite volume scheme helps to alleviate the accuracy problem for low-Mach flows. However this procedure is not fully satisfactory since lack of robustness is observed due to grid-dependent results. Note that the same remark holds for Discontinous Galerkin schemes[7].

To get further insight in the behavior of the ROE-FV-MLS scheme at very low Mach numbers, the normalized pressure fluctuations are computed as

$$p_{norm} = \frac{p_{max} - p_{min}}{p_{max}} \tag{3.15}$$

where p_{max} and p_{min} are the maximum and minimum pressures on the computational domain.

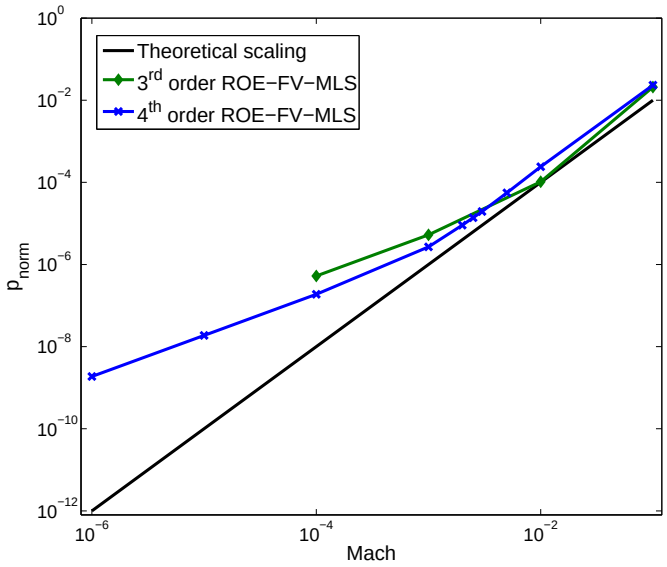
Fourth order ROE-FV-MLS method			
Mach	Mesh	C_{DRAG}	Order
10^{-1}	32×16	4.05×10^{-2}	-
	48×24	8.30×10^{-3}	3.91
	64×32	2.52×10^{-3}	4.15
	96×48	4.58×10^{-4}	4.20
10^{-2}	32×16	3.93×10^{-1}	-
	48×24	7.04×10^{-2}	4.24
	64×32	1.96×10^{-2}	4.44
	96×48	3.31×10^{-3}	4.39
10^{-3}	32×16	Non-physical	-
	48×24	Non-physical	-
	64×32	Non-physical	-
	96×48	3.11×10^{-2}	-

Table 3.1: Inviscid flow past a cylinder test case. Accuracy orders for the 4^{th} ROE-FV-MLS scheme for different Mach numbers. For $M_\infty = 10^{-3}$ we have only obtained a physical solution for the finest grid.

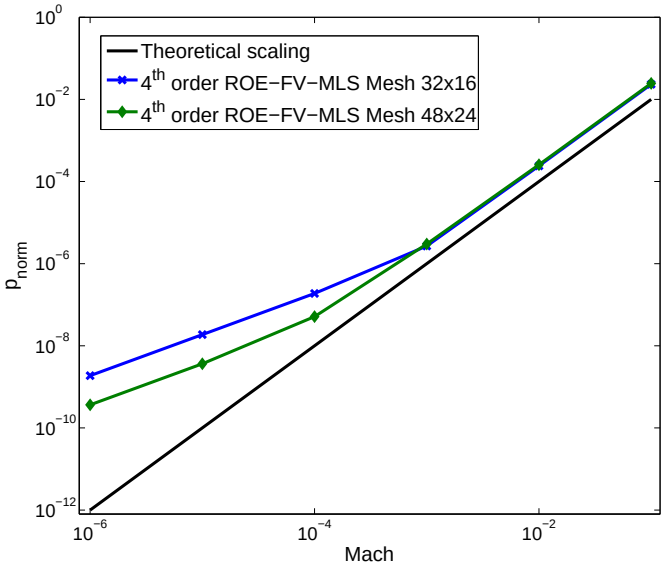
In figure 3.4 (a), it is observed that both 3^{rd} and 4^{th} ROE-FV-MLS numerical simulations, performed on the 32×16 grid, exhibit pressure fluctuations that are $\mathcal{O}(M_\infty^2)$ until a given Mach number.

The comparison in Figure 3.4 (b) of plots of the pressure fluctuations against M_∞ for two grid levels (namely the 32×16 and the 48×24 grids) clearly shows the grid dependence of the correct $\mathcal{O}(M_\infty^2)$ pressure scaling for a given Mach number.

In [185] it is shown that, for the inviscid low-Mach flow past a cylinder, the first order Roe scheme verifies a scaling with the Mach number of the form $N \sim M_\infty^{-1}$, where N is the number of points on the cylinder wall required to obtain a physical solution. We have performed a study of that scaling for the 4^{th} order FV-MLS scheme, and a scaling of the form $N \sim M_\infty^{-0.388}$ has been obtained. It is clear that increasing the order of the scheme decreases the Mach number of the flow that is possible to accurately compute on a given grid. Note that the scaling study has been performed considering that a solution



a)



b)

Figure 3.4: Inviscid flow past a cylinder test case. a)Pressure-Mach scaling for the 3^{rd} and 4^{th} order ROE-FV-MLS scheme in the 32×16 grid. b) Influence of the grid on the accuracy problem. Pressure-Mach scaling for the 4^{th} order ROE-FV-MLS scheme using different grids.

is accepted if the error in the $\mathcal{O}(M_\infty^2)$ scaling of the pressure fluctuations is smaller than a 10%.

To completely solve the accuracy problem we investigate the ability of combining

the ROE-FV-MLS scheme with the Rieper's fix presented in section 3.2.4. Although it was shown in [179] that this fix is very efficient when used with the first order Roe solver, to the author's knowledge, no results were published using second of higher-order reconstructions. Here, we perform an accuracy analysis by computing the error in the drag coefficient for $M_\infty = 10^{-1}$, $M_\infty = 10^{-2}$ and $M_\infty = 10^{-3}$. It can be clearly seen in Table 3.2 that the convergence orders are successfully recovered for all the Mach numbers considered. Contrary to the previous case, the use of the Rieper's fix allows physical solutions whatever the grid size, thus removing the grid-dependency problem.

		Third order		Fourth order	
Mach	Mesh	C_{DRAG}	Order	C_{DRAG}	Order
10^{-1}	32×16	1.56×10^{-2}	-	1.96×10^{-2}	-
	48×24	2.61×10^{-3}	4.41	2.92×10^{-3}	4.70
	64×32	1.05×10^{-3}	3.17	8.49×10^{-4}	4.30
	96×48	2.68×10^{-3}	3.37	1.69×10^{-4}	3.98
10^{-2}	32×16	7.88×10^{-3}	-	1.21×10^{-2}	-
	48×24	1.18×10^{-3}	4.69	1.63×10^{-3}	4.95
	64×32	5.61×10^{-4}	2.58	4.46×10^{-4}	4.50
	96×48	1.59×10^{-4}	3.11	8.84×10^{-5}	4.00
10^{-3}	32×16	6.48×10^{-3}	-	1.08×10^{-2}	-
	48×24	8.76×10^{-4}	4.94	1.36×10^{-3}	5.10
	64×32	4.50×10^{-4}	2.31	3.58×10^{-4}	4.65
	96×48	1.34×10^{-4}	2.99	7.05×10^{-5}	4.01

Table 3.2: Inviscid flow past a cylinder test case. Accuracy orders for the 3^{rd} and 4^{th} order ROE-FV-MLS scheme with Rieper's Fix for different Mach numbers.

As for the no fix case, the pressure scaling for low-Mach computations with the low Mach fix of Rieper is analyzed. Figure 3.5 shows that when the low-Mach fix is used, the correct $\mathcal{O}(M^2)$ scaling of the pressure fluctuations is recovered for all the computations.

Therefore we can conclude that using the low-Mach fix, the numerical viscosity of the high-order ROE-FV-MLS scheme is reduced to the right amount solving the accuracy problem.

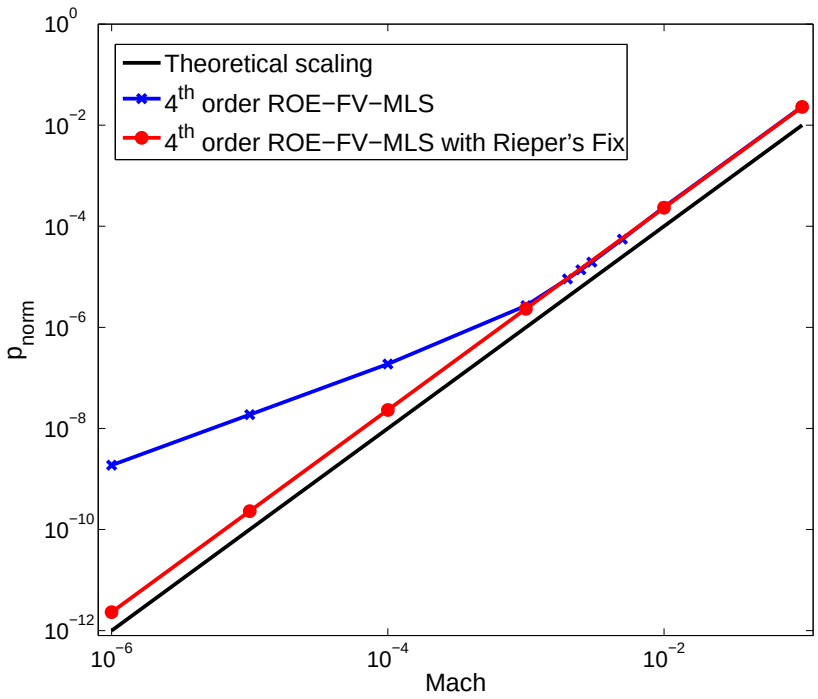


Figure 3.5: Inviscid flow past a cylinder test case. Pressure-Mach scaling for the 4^{th} order ROE-FV-MLS scheme with Rieper's Fix in the 32×16 grid.

As an illustration purpose, the pressure field obtained using a 4^{th} order ROE-FV-MLS scheme with Rieper's Fix at $M_\infty = 10^{-6}$ for the mesh 32×16 is plotted in Figure 3.6. Notice that the numerical solution, which is free from artificial wake downstream of the cylinder, presents a perfectly symmetric flow with respect to the coordinates axis.

Rusanov scheme

In this section, the previous study is extended to the FV-MLS solver with Rusanov flux (namely the RUS-FV-MLS method). As shown in Table 3.3, the 4^{th} order reconstruction scheme gives the expected order of convergence for Mach numbers $M_\infty = 10^{-1}$ and $M_\infty = 10^{-2}$. Note however that we were unable to obtain a physical solution for Mach numbers below 10^{-2}

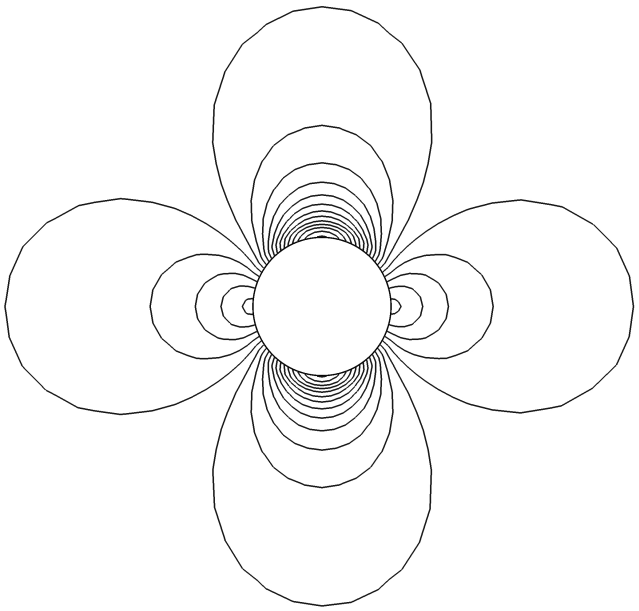


Figure 3.6: Inviscid flow past a cylinder test case. Pressure contours for $M_\infty = 10^{-6}$. The solution is obtained in the 32×16 grid by using the 4^{th} order ROE-FV-MLS scheme and the low-Mach fix of Rieper.

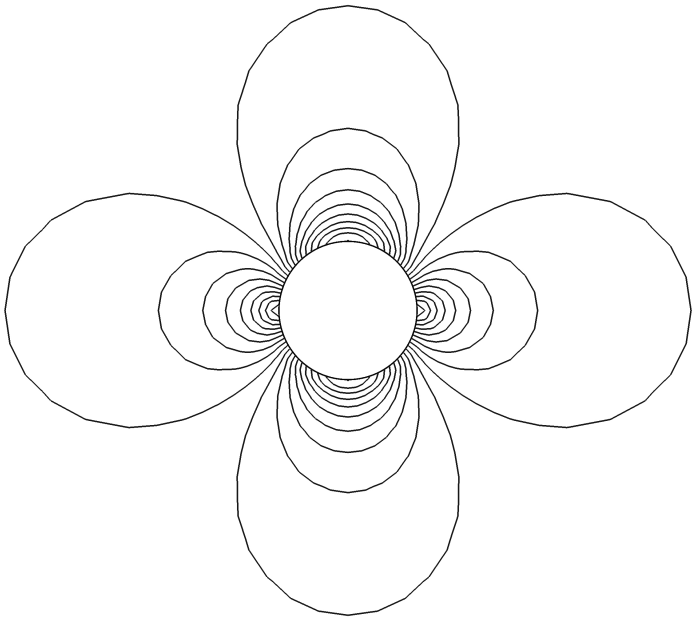


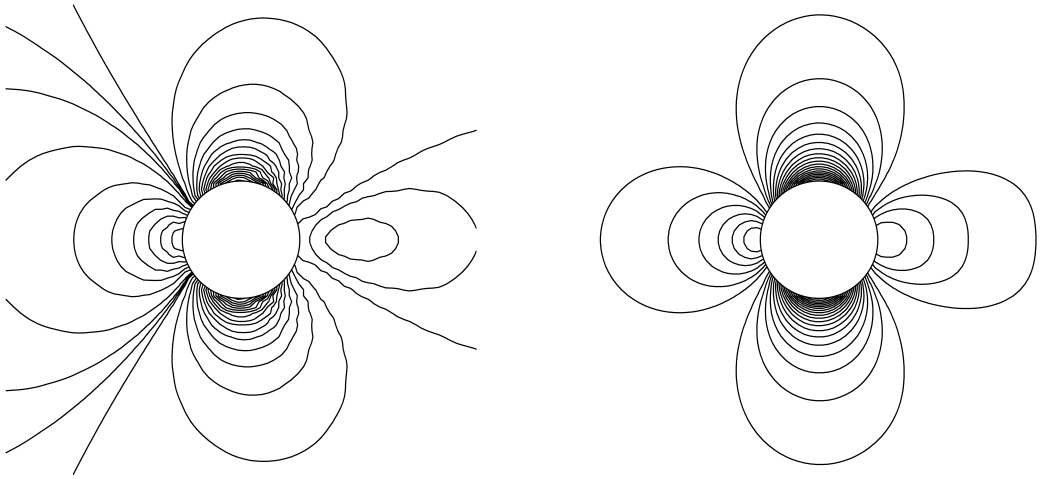
Figure 3.7: Inviscid flow past a cylinder test case. Mach contours for $M_\infty = 10^{-6}$. The solution is obtained in the 32×16 grid by using the 4^{th} order ROE-FV-MLS scheme and the low-Mach fix of Rieper.

Fourth order RUS-FV-MLS method			
Mach	Mesh	C_{DRAG}	Order
10^{-1}	32×16	2.36×10^{-1}	-
	48×24	4.85×10^{-2}	3.90
	64×32	1.41×10^{-2}	4.30
	96×48	2.09×10^{-3}	4.71
10^{-2}	32×16	1.04×10^{0}	-
	48×24	3.49×10^{-1}	2.70
	64×32	1.24×10^{-1}	3.59
	96×48	2.14×10^{-2}	4.34
10^{-3}	32×16	Non-physical	-
	48×24	Non-physical	-
	64×32	Non-physical	-
	96×48	Non-physical	-

Table 3.3: Inviscid flow past a cylinder test case. Accuracy orders for the 4^{th} order RUS-FV-MLS scheme for different Mach numbers.

Next, results of the RUS-FV-MLS scheme combined with the Li and Gu's low-Mach fix presented in section 3.2.4 is discussed in detail. First, it is important to note that, for 1^{st} order HLL schemes, the low-Mach fix does not solve completely the checkerboard problem [117] as shown by the pressure contours for $M_\infty = 10^{-2}$ obtained on a 96×48 grid (Fig. 3.8 (a)). On the contrary, the solutions of the 4^{th} order RUS-FV-MLS scheme with Li and Gu's fix is free from checkerboard. This is due to the reduction on the density difference $\Delta(\rho)$ term (related with the pressure difference term [117]) in the continuity equation when the order is increased, as seen in Figure 3.1. Although the 4^{th} high-order scheme does not give a fully symmetrical solution, it can be observed that the solution is greatly improved compared to its low order counterpart.

Table 3.4 shows that the use of the Li and Gu's fix for the Rusanov scheme allows to recover the expected order of accuracy for all the tested Mach numbers. Recall that, with the grids considered, it was not possible to obtain a physically solution for a Mach



a) b)

Figure 3.8: Inviscid flow past a cylinder test case. Pressure contours for $M_\infty = 10^{-2}$. The solution is obtained in the 96×48 grid by using the Li and Gu's low-Mach fix with a first order FV Rusanov scheme (a) and the 4^{th} order RUS-FV-MLS scheme. It is observed that the first order scheme presents a weak checkerboard that is removed with the use of the high-order scheme.

number below $M_\infty = 10^{-2}$ without using the fix (Table 3.3).

In Figure 3.9 it is observed that the 4^{th} order FV-MLS scheme with the Rusanov numerical flux loses the right scaling of the pressure ($\mathcal{O}(M_\infty^2)$) at a greater value of the Mach number than that obtained when the Roe flux is used, Figure 3.5. This is an expected result, since the Rusanov scheme is an asymptotically inconsistent scheme. Note however that the use of the Li and Gu's fix gives the correct pressure scaling.

3.4 Low-Mach fixes and limiters

In compressible flows, it is common the presence of shocks, that is, the presence of discontinuities in the solution. The basic high-order Finite Volume method does not respect the Total Variation Diminishing (TVD) property and can therefore lead to oscillatory solutions when discontinuous flow fields are computed. In 1959, Godunov [65] showed that it is not possible for a linear schemes of second or higher order of accuracy to be both higher than first order accurate and free of spurious oscillations.

Fourth order Rusanov scheme with Li and Gu's Fix			
Mach	Mesh	C_{DRAG}	Order
10^{-1}	32×16	6.53×10^{-2}	-
	48×24	1.04×10^{-2}	4.54
	64×32	2.84×10^{-3}	4.51
	96×48	4.40×10^{-4}	4.59
10^{-2}	32×16	5.31×10^{-2}	-
	48×24	8.46×10^{-3}	4.53
	64×32	2.30×10^{-3}	4.52
	96×48	3.47×10^{-4}	4.67
10^{-3}	32×16	5.15×10^{-2}	-
	48×24	8.14×10^{-3}	4.55
	64×32	2.20×10^{-3}	4.55
	96×48	3.41×10^{-4}	4.58

Table 3.4: Inviscid flow past a cylinder test case. Accuracy orders for the 4^{th} order RUS-FV-MLS scheme with Li and Gu's fix for different Mach numbers.

3.4.1 Slope limiters

Slope limiters are very common numerical technique used in the MUSCL framework formulations to recover the TVD property.

In Figure 3.10 a), it is shown a schematic representation of a discontinuity at j. In order to obtain a high-accurate discretization of the convective fluxes of equation (3.2), the reconstructed variables $\boldsymbol{U}_j^+$ and $\boldsymbol{U}_j^+$ are needed. For a steady state test case, a second-order reconstruction of the variable can be computed as

$$\boldsymbol{U}_j^+ = \boldsymbol{U}_I + \boldsymbol{\nabla U}_I \cdot (\boldsymbol{x}_j - \boldsymbol{x}_I) \tag{3.16}$$

where the gradient $\boldsymbol{\nabla U}_I$ is computed at cell centroid I. As seen in Figure 3.10 a), the presence of a discontinuity introduces a non-realistic reconstruction, that will lead to spurious oscillations in the solution. The idea behind the slope limiters, is to create a limited higher-order (piecewise linear discontinuous) reconstruction of the solution with a limited

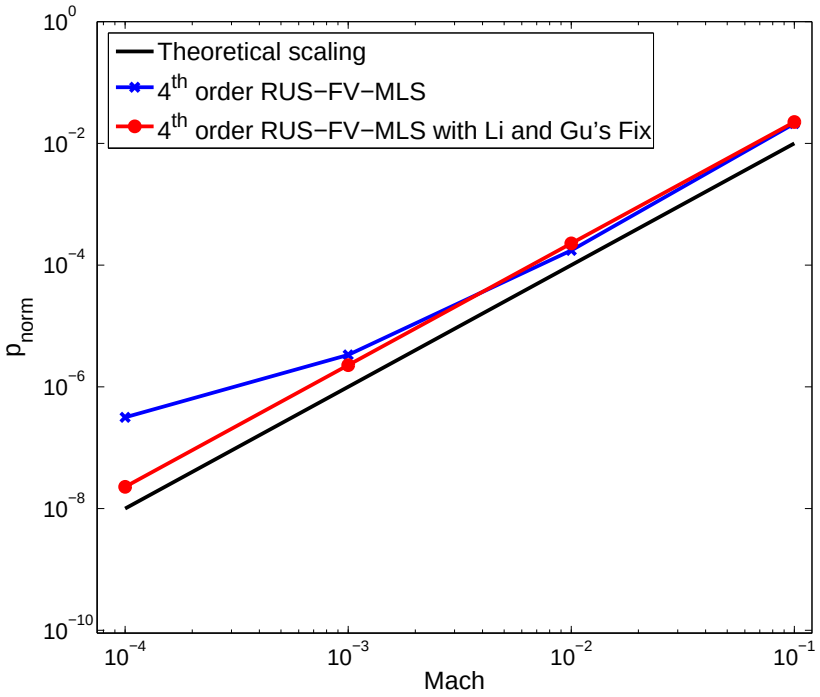


Figure 3.9: Inviscid flow past a cylinder test case. Comparison of the pressure-Mach scaling for the 4^{th} order RUS-FV-MLS scheme with and without the Li and Gu's Fix for the 32×16 grid.

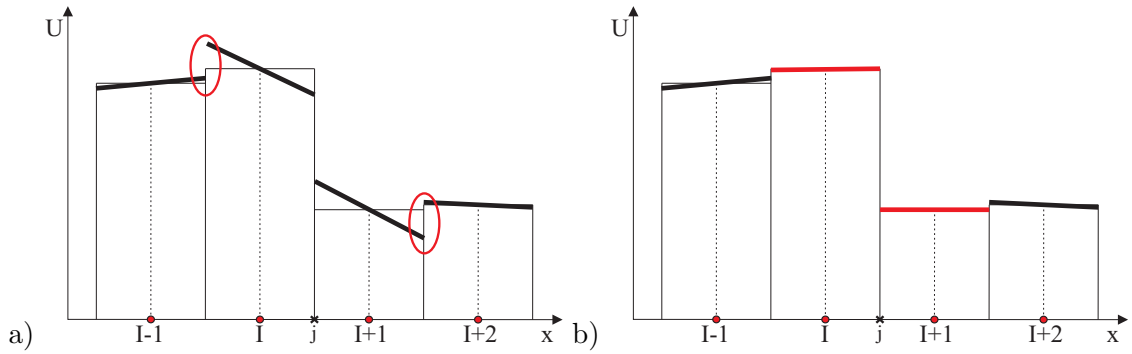


Figure 3.10: Reconstructed solution using a) unlimited piecewise linear and b) limited piecewise linear reconstruction.

gradient that enforces the TVD condition, as seen in Figure 3.10 b).

In this Chapter three slope limiters are studied: Barth and Jespersen [3], Van Albada [210] and Venkatakrishnan [217] slope limiters.

Barth and Jespersen slope limiter

Barth and Jespersen [3] introduced the first multidimensional limiter which allowed oscillation-free solutions of transonic flows on irregular triangular meshes. The limiter presented by Barth-Jespersen has been widely used [14].

For example, the second-order reconstruction can be expressed as

$$\boldsymbol{U}_j^+ = \boldsymbol{U}_I + \chi_I \boldsymbol{\nabla U}_I \cdot (\boldsymbol{x}_j - \boldsymbol{x}_I) \tag{3.17}$$

where χ_I is defined as the slope limiter, with a range $0 \leq \chi_I \leq 1$. Therefore, when $\chi_I = 0$ a first-order scheme is obtained, and when $\chi_I = 1$ the limiter is deactivated and the full second-order reconstruction is employed. For a higher-order reconstruction the reconstruction is expressed as

$$\boldsymbol{U}_j^+ = \boldsymbol{U}_I + \chi_I \left(\boldsymbol{\nabla U}_I \cdot (\boldsymbol{x}_j - \boldsymbol{x}_I) + \frac{1}{2} (\boldsymbol{x}_j - \boldsymbol{x}_I)^T \boldsymbol{H} \, (\boldsymbol{x}_j - \boldsymbol{x}_I) + ... \right) \tag{3.18}$$

The following procedure is used by Barth-Jespersen for the computation of the slope limiter χ_I:

1. Find the largest negative ($\delta \boldsymbol{U}_I^{min} = \min(\boldsymbol{U}_\mathrm{I} - \boldsymbol{U}_\mathrm{k})$) and positive ($\delta \boldsymbol{U}_I^{max} = \max(\boldsymbol{U}_\mathrm{I} - \boldsymbol{U}_\mathrm{k})$) difference between the solution in the immediate neighbors and the current control volume.

2. Compute the unconstrained reconstructed value at each Gauss point ($\boldsymbol{U}_j$).

3. Compute the maximum allowable value of χ_{Ij} for each Gauss point j

$$\chi_{Ij} = \begin{cases} \min\left(1, \frac{\delta \boldsymbol{U}_\mathrm{I}^{\max}}{\boldsymbol{U}_\mathrm{j} - \boldsymbol{U}_\mathrm{I}}\right) & , \boldsymbol{U}_j - \boldsymbol{U}_I > 0 \\ \min\left(1, \frac{\delta \boldsymbol{U}_\mathrm{I}^{\min}}{\boldsymbol{U}_\mathrm{j} - \boldsymbol{U}_\mathrm{I}}\right) & , \boldsymbol{U}_j - \boldsymbol{U}_I < 0 \\ 1 & , \boldsymbol{U}_j - \boldsymbol{U}_I = 0 \end{cases} \tag{3.19}$$

4. Select $\chi_I = \min(\chi_\mathrm{Ij})$

Venkatakrishnan slope limiter

Steps 1,3 and 4 of the Barth-Jespersen slope limiter introduce non-differentiability in the computation of the reconstructed function [133]. Venkatakrishnan [217] presented a smooth alternative to step 3 of the Barth-Jespersen slope limiter. As pointed in [133], in practice the non-differentiability of step 3 causes the greatest degradation in convergence performance. The modification proposed by Venkatakrishnan [217] reads:

$$\chi_{Ij} = \begin{cases} \frac{1}{\Gamma}\left[\frac{(\Delta_+^2+\epsilon^2)\Gamma+2\Gamma^2\Delta_+}{\Delta_+^2+2\Gamma^2+\Gamma\Delta_++\epsilon^2}\right] & ,\boldsymbol{U}_j-\boldsymbol{U}_I>0 \\ \frac{1}{\Gamma}\left[\frac{(\Delta_-^2+\epsilon^2)\Gamma+2\Gamma^2\Delta_-}{\Delta_-^2+2\Gamma^2+\Gamma\Delta_-+\epsilon^2}\right] & ,\boldsymbol{U}_j-\boldsymbol{U}_I<0 \\ 1 & ,\boldsymbol{U}_j-\boldsymbol{U}_I=0 \end{cases} \tag{3.20}$$

where $\Gamma = \boldsymbol{U}_j - \boldsymbol{U}_I$, $\Delta_+ = \delta\boldsymbol{U}_I^{max}$, $\Delta_- = \delta\boldsymbol{U}_I^{min}$ and ϵ is a parameter introduced to avoid the unnecessary activation of the limiter in regions of nearly uniform flow, and it is defined in the two-dimensional case as $\epsilon^2 = (K\ \Delta x)^2$, where K is a tunable parameter and Δx is a characteristic length for the control volume I. The choice of the parameter K is a compromise. Large values of K are favorable to accuracy in smooth regions and good convergence. However, since for any $K > 0$ the limiter no longer strictly enforces monotonicity, large values of K can lead to significant overshoots near discontinuities in the solution. In this thesis the parameter $K = 5$ is chosen, as recommended in [16].

Van Albada limiter

Van Albada [210] presented a general strategy to obtain limited gradients and successive derivatives, needed for the high-order reconstruction. Van Rosendale [216] has proposed a multidimensional generalization of the Van Albada limiter for unstructured triangular meshes. Later, Cueto-Felgueroso et al. [39] extended the formulation for unstructured quadrilateral cells.

The averaged gradient of cell I, $\boldsymbol{\nabla U}_I$, is obtained as:

$$\boldsymbol{\nabla U}_I = \sum_{k=1}^{N} \omega_k \boldsymbol{\nabla U}_k \tag{3.21}$$

where $\boldsymbol{\nabla U}_k$ represents the neighboring gradients, and the weight, ω_k, is obtained according

to [39, 93] as

$$\omega_k = \frac{\prod_{i \neq k}^{N} g_i + \epsilon^{N-1}}{\sum_{j=1}^{N} \left(\prod_{i \neq j}^{N} g_i \right) + N\epsilon^{N-1}} \tag{3.22}$$

where $\boldsymbol{g}$ is function of the unlimited gradients. As propossed in [39] in this thesis $g_i = \|\boldsymbol{\nabla U}_i\|^2$, and $\epsilon = 10^{-10}$ to avoid division by zero.

The same procedure is used to obtain the successive derivatives needed for the high-order reconstruction.

3.4.2 MLS-based shock sensor

Most of the slope limiters were developed for second-order schemes, and according to [151] the extension proposed in equation (3.18) for higher-order schemes presents problems, and therefore the use of higher-order schemes with limiters does not guarantee a more accurate solution. One possible improvement is the use of selective limiting techniques, that are based on the principle of activate the limiter only where it is necessary. In this context, the MLS-based shock-wave sensor was presented in [151]. It is based on the multiresolution properties of the Moving Least Squares approximations [125]. The basic idea of the MLS-sensor is to use these properties to separate the high scale components of the solution in order to develop an MLS-based wavelet function of the density that acts as the reference variable.

Following [151], the slope limiter algorithm is activated when the following condition is verified:

$$\left| \sum_{k=1}^{n_I} \rho_k ({N_k}^{s_x^H}(\boldsymbol{x}) - {N_k}^{s_x^L}(\boldsymbol{x})) \right| > T_v \tag{3.23}$$

Setting $s_x^H = 2s_x^L$ the term $\sum_{k=1}^{n_I} \rho_k ({N_k}^{s_x^H}(\boldsymbol{x}) - {N_k}^{s_x^L}(\boldsymbol{x}))$ represents the high-scale part of the density solution. The high-scale part has a greater value in the vicinity of shock waves. The threshold value T_v is a problem-dependent parameter, defined as

$$T_v = C_{lc} \left| \nabla \rho \right|_I (\Omega_I)^{\frac{1}{d}} / \max(M_l) \tag{3.24}$$

where Ω_I is the size (area in 2D) of the control volume I, d is the number of dimensions of the problem, $C_{lc} = 0.32$ is a case-dependent parameter and $\max(M_l)$ is the maximum local Mach number in the computational domain. If the parameter would be chosen as $C_{lc} = 0$ the slope-limiter algorithm will be activated in the whole domain of computation.

3.4.3 Roe scheme with Rieper's fix

A high-order all-speed scheme must include a limiting mechanism in order to be Total Variation Diminishing (TVD) for supersonic and transonic cases. Ideally, the limiter should not be active in smooth or low Mach regions, in order to keep the accuracy of the high-order scheme. When Rieper's fix is used the limiter may introduce spurious pressure oscillations that spoil the solution. To show this, the inviscid flow past a circular cylinder is computed at $M_\infty = 10^{-3}$. The unstructured mesh shown in Figure 3.11 has 64 elements on the cylinder surface and a total number of 2320 elements.

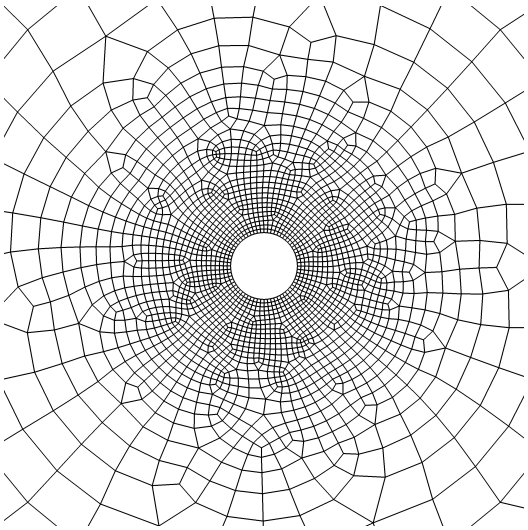


Figure 3.11: Close view of the unstructured O-grid employed for the computation of the inviscid flow past a circular cylinder test case using slope limiters.

Figures 3.12 a) and b) present the pressure contours obtained by combining the Barth and Jespersen [3] and the Van Albada [210] non-differentiable limiters with the Rieper's fix.

It is observed that the accuracy of the fix is clearly affected by the use of these limiting strategies. As suggested in [176], Figure 3.12 *c*) shows that this effect is avoided using a differentiable limiter such as the Venkatakrishnan slope limiter [217]. However, Venkatakrishnan limiter may exhibit deviations from the monotone solution near strong shocks [218]. For these problems we may prefer to use a non-differentiable limiter that could introduce spurious pressure oscillations. This problem is avoided using a shock-sensor. In this thesis, it is proposed to employ an MLS-based sensor proposed in [151] for the Euler equations and explained in section 3.4.2. The MLS sensor avoids that the slope limiter is activated in smooth regions, thus eliminating the appearance of spurious

pressure oscillations. Such behavior is clearly observed in Figure 3.12 *d*) obtained using the MLS sensor based selective application of the Van Albada limiter. Note that computations performed using the MLS sensor with the limiter of Barth and Jespersen (not shown) lead to similar conclusions.

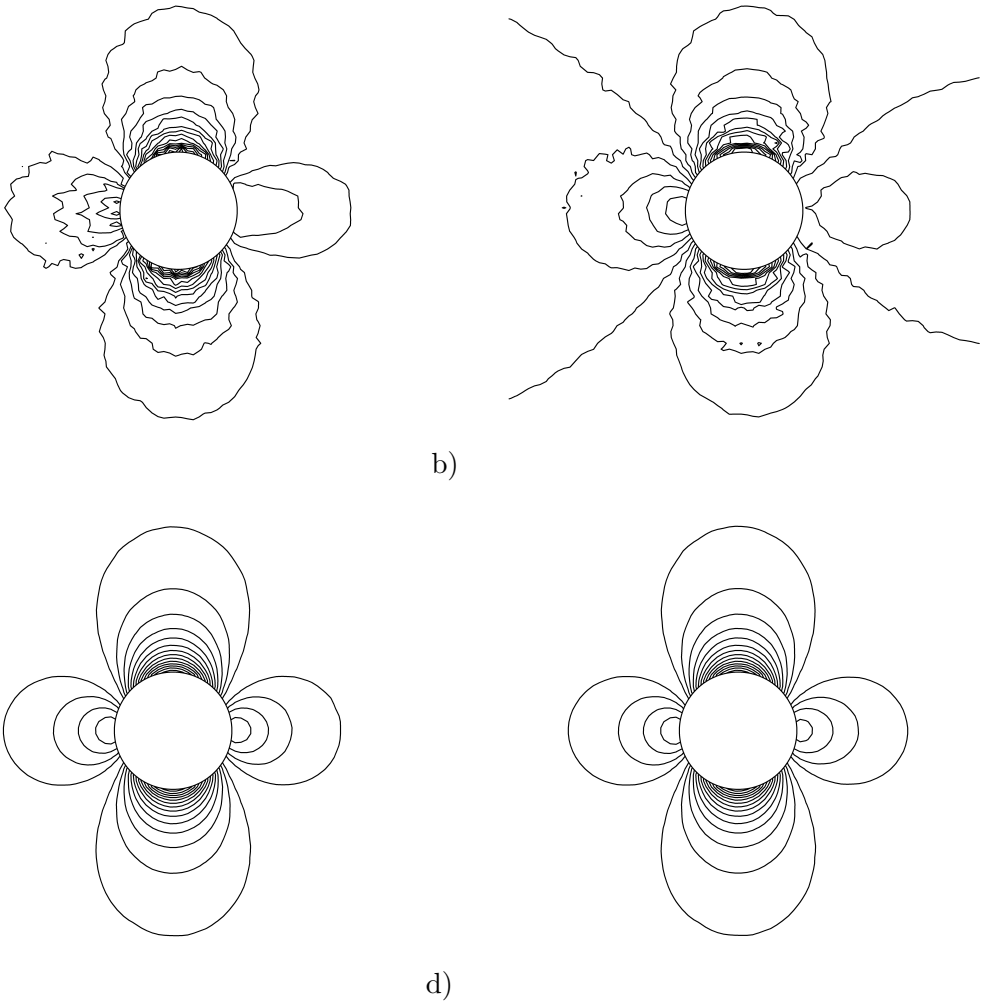


Figure 3.12: Inviscid $M_\infty = 10^{-3}$ flow past a cylinder on an unstructured grid. Pressure contours. Rieper's low-Mach fix with non-differentiable limiters: Barth-Jespersen limiter (a), Van Albada limiter (b). Rieper's low-Mach fix with Venkatakrishnan limiter (c) and with Van Albada limiter and MLS-based sensor (d).

Table 3.5 summarizes the drag coefficient obtained with the different slope limiters. While the solutions corresponding to the Barth-Jespersen and Van-Albada slope limiters are far from the solution obtained without slope limiter, the solutions obtained with Van-Albada limiter with MLS-based sensor and Venkatakrishnan slope limiter are similar. The

difference between the reference solution and the solution obtained with Venkatakrishnan's limiter is due to the unnecessarily activation of the slope limiter.

Limiter	C_{DRAG}
Barth-Jespersen [3]	8.64×10^{-2}
Van Albada [210]	2.37×10^{-1}
Venkatakrishnan [217]	3.43×10^{-3}
Van Albada + MLS-based sensor	3.34×10^{-3}
No limiter	3.34×10^{-3}

Table 3.5: Inviscid $M_\infty = 10^{-3}$ flow past a cylinder on an unstructured grid. Comparison of the drag coefficient for different slope limiters. The solutions are obtained with a fourth order Rieper's Fix Roe FV-MLS scheme on an unstructured mesh.

3.4.4 Rusanov scheme with Li and Gu's fix

The application of slope limiters to the low-Mach fix of Li and Gu produces the same problems than those reported in previous sections for the Rieper's fix.

In Figure 3.13 a) and b) we show that the use of non-differentiable limiters of Barth-Jespersen and Van Albada with the low Mach fix gives spurious pressure oscillations. As in the Roe's case, the use of a differentiable limiter or the MLS-based sensor solves the problem, as it is shown in Figure 3.13 c) and Figure 3.13 d).

3.5 3D decay of compressible isotropic turbulence

In this example, it is investigated the ability of the high-order FV-MLS formulation for all-speed to solve turbulent flows. To this end, the problem of the compressible isotropic turbulence decay is analyzed. This test case corresponds to case 6 defined in [193], and it is widely used as a benchmark for subgrid scale models for Large-Eddy Simulation (LES) [90].

The three-dimensional computational domain is a cube of size 2π with periodic boundary conditions. The computational domain, $\Omega_T = [-\pi, \pi]^3$, is discretized using a structured mesh of $32 \times 32 \times 32$ elements.

In this example, following [152] an Implicit Large-Eddy Simulation (ILES) approach is used, so no explicit turbulence model has been used for the computations. The dissipation introduced by the numerical scheme plays the role of the subgrid scale model.

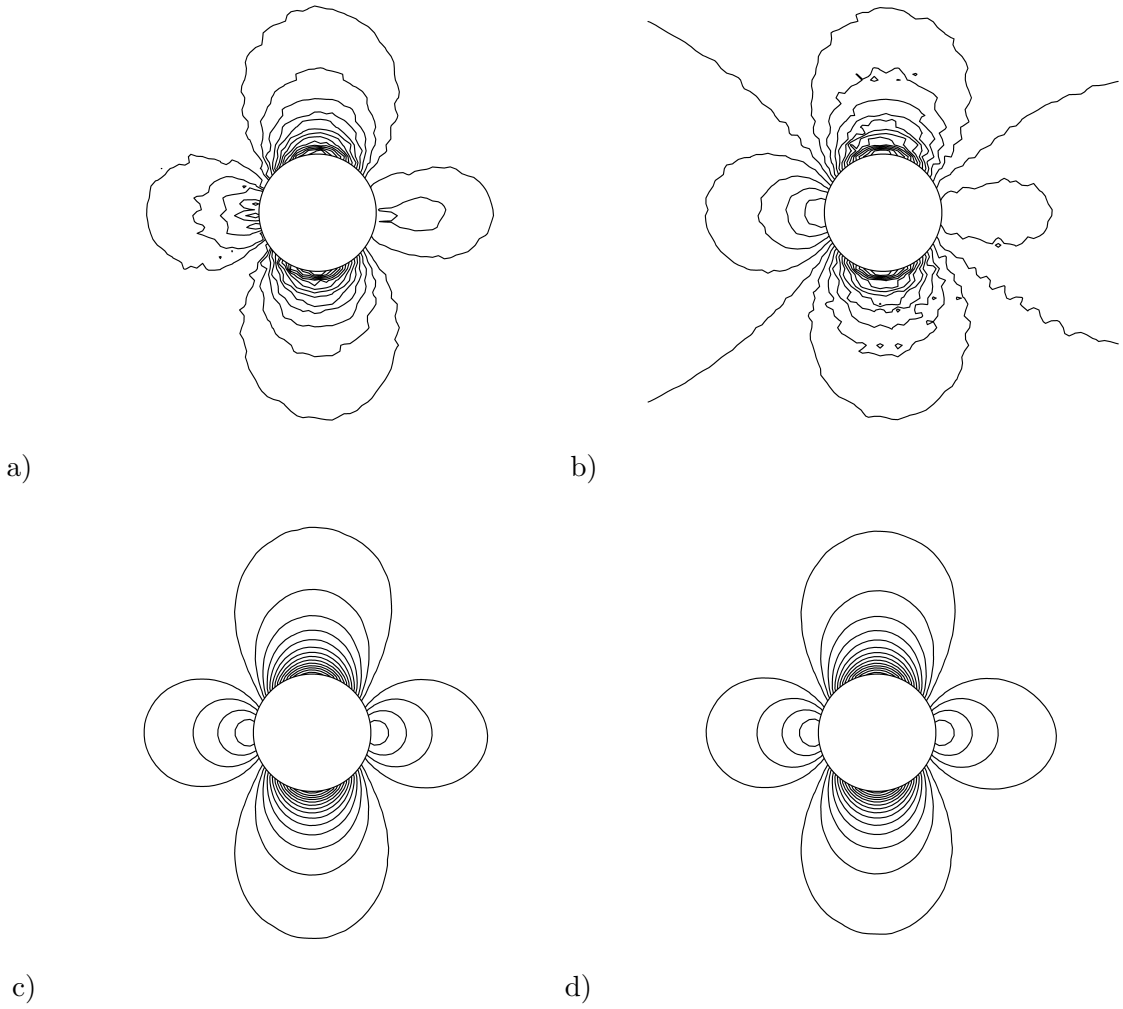


Figure 3.13: Inviscid $M_\infty = 10^{-3}$ flow past a cylinder on an unstructured grid. Pressure contours. Rusanov's low-Mach fix with non-differentiable limiters: Barth-Jespersen limiter (a), Van Albada limiter (b). Rieper's low-Mach fix with Venkatakrishnan limiter (c) and with Van Albada limiter and MLS-based sensor (d).

The aim of this numerical example is to study the behavior of the all-speed scheme developed in this Chapter on turbulent flows. Notice that the fix reduces the dissipation within the Riemann solver in order to improve the solution at low Mach numbers. A similar study has been carried out by Thornber in [202].

As a reference solution, the turbulence decay is computed using a sixth-order compact finite difference scheme on a $128 \times 128 \times 128$ elements with a LES subgrid model model. The result of the reference LES computation coincides with the DNS solution given in [193].The time step is $\Delta t = 0.05$ for the coarsest grid (32^3 elements) and $\Delta t = 0.01$

for the reference solution. These time steps corresponds to approximately to 250 and 1250 time-steps per eddy turnover time (τ_0). The eddy turnover time is defined as the ratio of the turbulent kinetic energy to the dissipation rate based on the initial field.

Figure 3.14 shows the decay of the turbulent kinetic energy for the DNS and the third-order ROE-FV-MLS scheme with Rieper's fix and the Van Albada limiter with the MLS-based sensor.

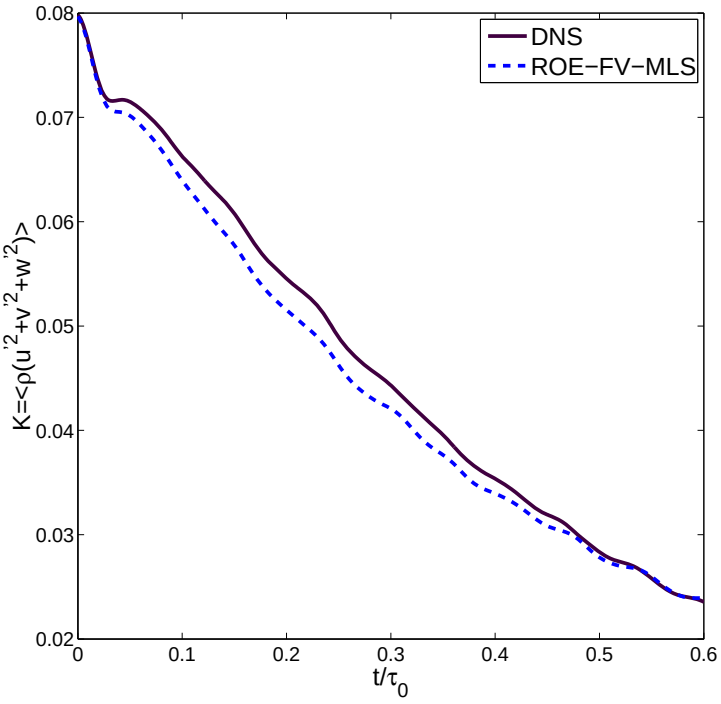


Figure 3.14: Time history of the turbulent kinetic energy decay. The solution is obtained with 3^{rd} ROE-FV-MLS with Rieper's Fix on a structured mesh with 32^3 elements.

The instantaneous three-dimensional energy spectrum at time $t/\tau_0 = 0.3$ is plotted in Figure 3.15. The solution capture the two different slopes predicted by the Eddy-Damped Quasi-Normal Markovian Theory (EDQNM) [83].

3.6 Unsteady transonic viscous flow over a circular cylinder

This last example aims to highlight the robustness and accuracy capabilities of the all-speed FV-MLS formulation. To this end, the transonic viscous flow past a circular cylinder is computed at free-stream Mach number equal to 0.8 involving complex viscous-

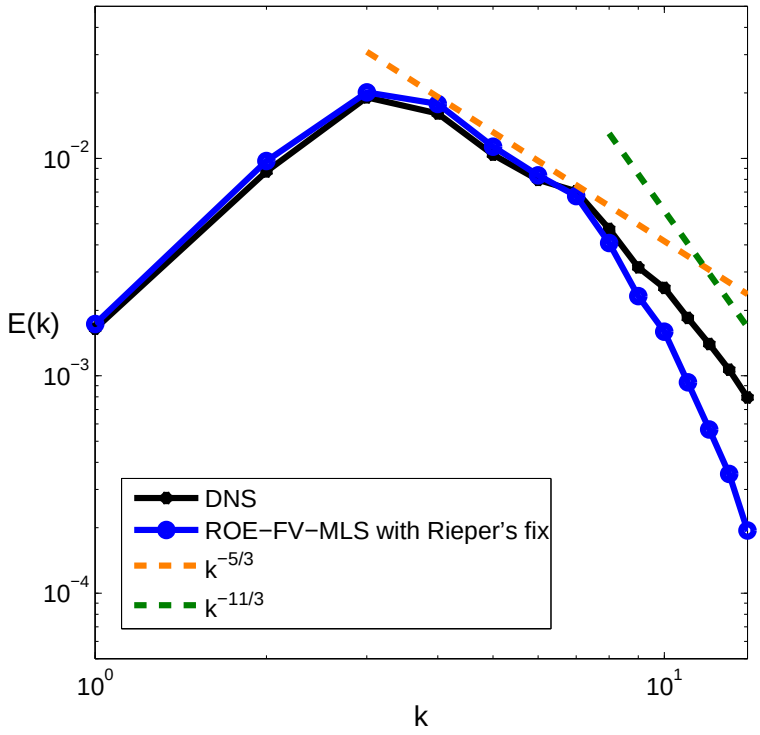


Figure 3.15: Instantaneous three-dimensional energy spectrum at $t/\tau_0 = 0.3$. The solution is obtained with 3^{rd} ROE-FV-MLS with Rieper's Fix on a structured mesh with 32^3 elements.

shock interactions and vortex shedding in the wake of the cylinder[142].

The computational setup considered here was taken from reference [63]. The Reynolds number based on a diameter of the cylinder $D = 1$ m is $Re = 166.000$.

The compressible Navier-Stokes equations are solved in a circular region with the 3^{rd} order ROE-FV-MLS method using the Rieper's Fix. Three different limiting techniques has been studied: the Van Albada limiter, the Venkatakrishnan limiter and the MLS sensor with the Van Albada limiter. The outer boundary of the O-topology mesh is located at a distance of 200 diameters from the center of the cylinder. The cylinder surface is discretized with 720 control volumes. The normal distance of the first centroid to the cylinder wall is $y_n = 2.85 \times 10^{-4} D$. The total number of control volumes is 206.150.

The FV-MLS numerical results are compared with the experimental results presented in [142] and with the numerical results obtained using 2D Detached Eddy Simulation (DES) based on a 4^{th} order finite volume method[63].

In order to show the complex viscous-shock interaction we plot in Figure 3.16 the

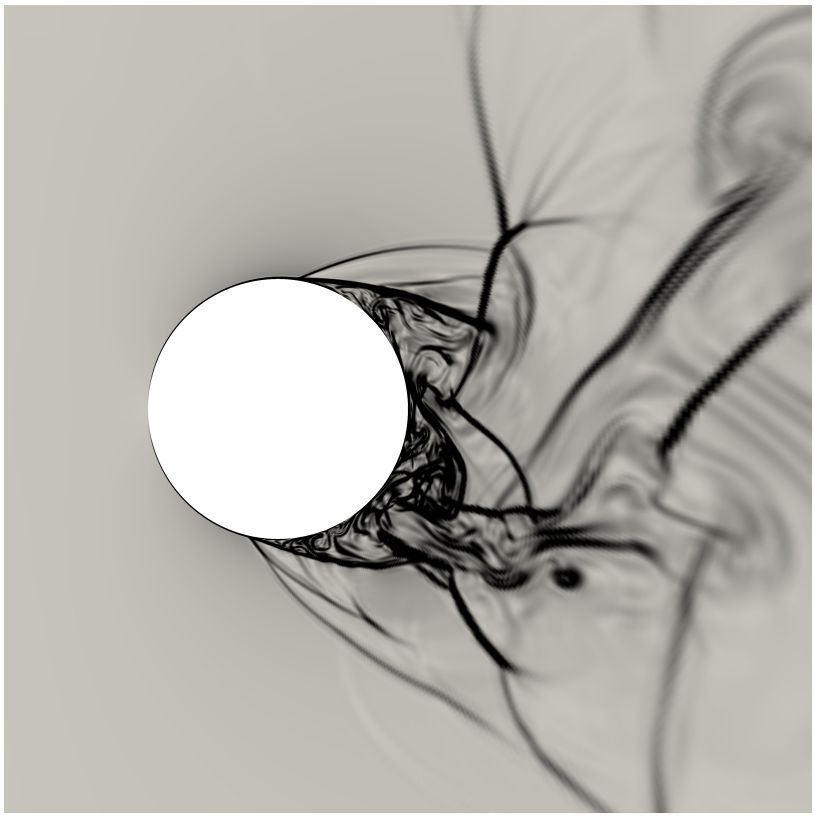


Figure 3.16: Unsteady transonic viscous flow over a circular cylinder. Magnitude of the temperature gradient near the cylinder for $t^* = 49.93$. The solution is obtained with 3^{rd} ROE-FV-MLS with Rieper's Fix and Van Albada limiter with MLS-based sensor.

magnitude of the temperature gradient, as suggested in [63].

Figure 3.17 shows the Mach field for a non-dimensional time based on the inlet velocity and the diameter of the cylinder ($t^* = tU_{inlet}/D = 49.93$), as suggested in [63] . It is seen that a von Kármán vortex street is formed after the viscous-shock interaction area.

Figure 3.18 shows the pressure contours and the control volumes where the MLS-based sensor is activated. It is seen than the limiter is not activated on a great part of the low-mach region thus preserving the accuracy of the flow structure.

Figure 3.19 present a comparison of the distribution of the pressure coefficient over the cylinder between the FV-MLS results with the different limiting techniques and the experimental data from [142]. It is seen that the mean position of the boundary layer separation point is well predicted (around 75 degrees) when the MLS-based shock sensor is used in conjunction with the Van Albada slope limiter. The solution obtained by the FV-MLS with the Van Albada slope limiter presents a mean detachment of the boundary

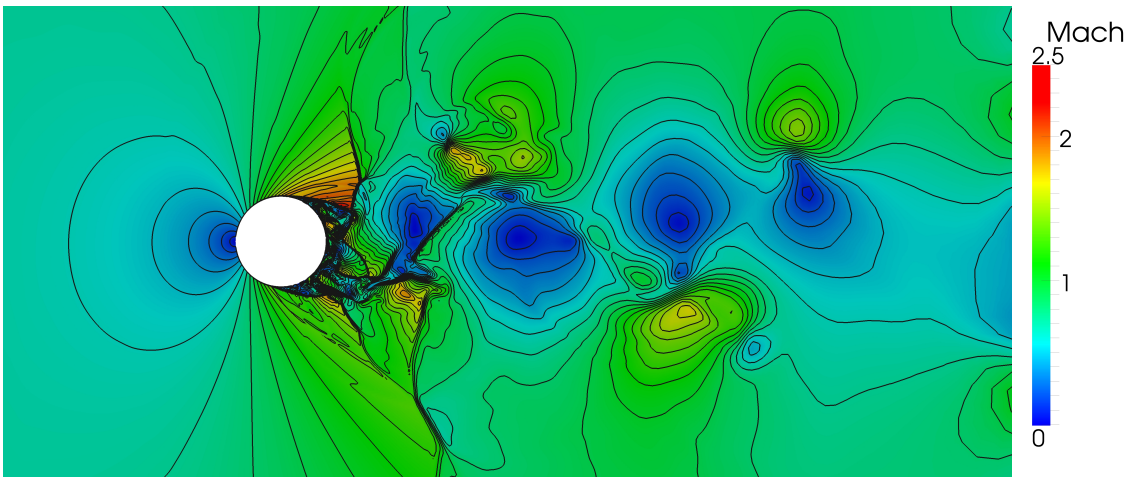


Figure 3.17: Unsteady transonic viscous flow over a circular cylinder. Mach number field for $t^* = 49.93$. The solution is obtained with 3^{rd} ROE-FV-MLS with Rieper's Fix and Van Albada limiter with MLS-based sensor.

at around 60 degrees. The solution is improved when the Venkatakrishnan slope limiter is applied. Note that the results presented in this work are obtained using a third-order method, whereas the reference solution is computed with a fourth-order method.

The discrepancies observed with the experimental results after the boundary layer separation point can be originated by the two-dimensional character of the present computations. Note that similar results were obtained in [63].

The examination of the average drag coefficient presented in Table 3.6 confirms the previous analysis. Although the computed drag coefficient computed using the FV-MLS approach agrees to those given in [63], there is not a good agreement with the experimental data[142] due to the 2D character of the present computations.

3.7 Conclusions

In this Chapter a high-order density-based finite volume compressible formulation for all-speed flows is presented. In order to solve the accuracy problem, two approaches has been addressed:

First, the order of the numerical scheme has been increased, which alleviates the accuracy problem. However, it is not completely solved and a grid dependency with the Mach number still remains. It has been shown that this dependence is smaller than for the first-order numerical scheme.

Second, the low-Mach fix proposed by Rieper for the first-order Roe scheme and

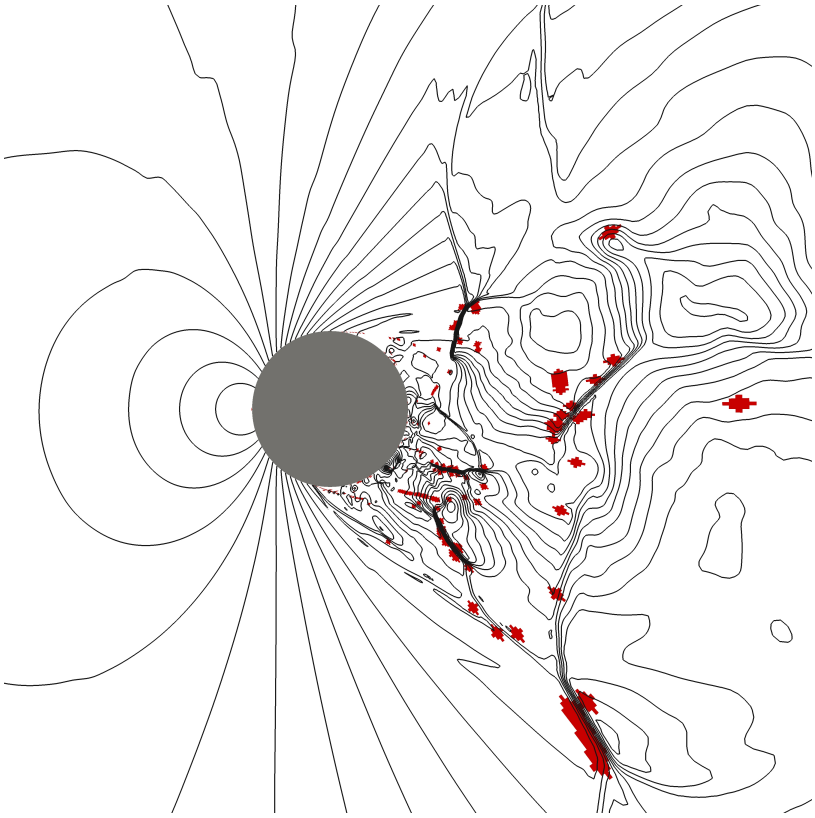


Figure 3.18: Unsteady transonic viscous flow over a circular cylinder. Pressure field and selective limiting for $t^* = 49.93$. Shaded cells indicates the elements where the slope limiter is activated. The solution is obtained with 3^{rd} ROE-FV-MLS with Rieper's Fix and Van Albada limiter with MLS-based sensor.

Method	C_{DRAG}
Reference 2D computations [63]	1.86
FV-MLS Van Albada	1.82
FV-MLS Venkatakrishnan	1.84
FV-MLS Van Albada+MLS-based sensor	1.81
Experimental [142]	1.50

Table 3.6: Unsteady transonic viscous flow over a circular cylinder. Comparison of the drag coefficient with other numerical and experimental results. The 2D ROE-FV-MLS solution is obtained with a 3^{rd} order Roe FV-MLS method with Rieper's Fix.

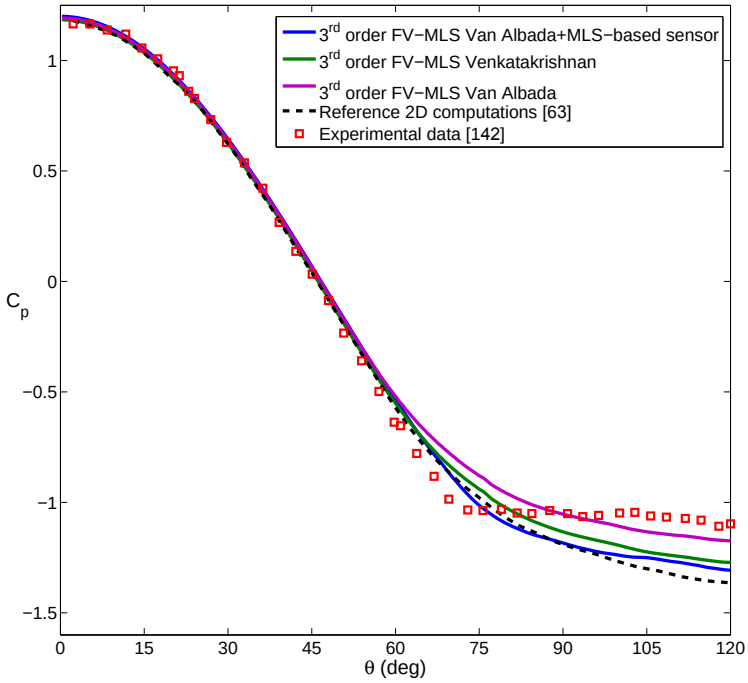


Figure 3.19: Unsteady transonic viscous flow over a circular cylinder. Surface pressure coefficient around the cylinder. The experimental data is taken from [142].

the the fix proposed Li and Gu for the HLL schemes, completely solve the accuracy problem.

A high-order all-speed scheme must include a limiting mechanism in order to be Total Variation Diminishing (TVD) for supersonic and transonic cases. Ideally, the limiter should not be active in smooth or low Mach regions, in order to keep the accuracy of the high-order scheme. However, both fixes produces erroneous pressure oscillations with non-differentiable limiters such as those of Barth and Jespersen or Van Albada. In non homogeneous grids even differentiable limiters such as Venkatakrishnan's limiter can be unnecessarily activated in low-Mach regions, producing a loss of accuracy. In order to avoid the unnecessarily activation of the limiter in low-Mach regions, the use of an MLS-based shock wave sensor is proposed.

The proposed high-order scheme is validated in several steady and unsteady test cases, where the robustness, accuracy and efficiency of the numerical method is confirmed.

Several authors [59, 182, 220] have placed the incompressibility limit at $M < 0.3$. As it was mentioned in Chapter 1, at this flow speed the maximum relative change in the density field is around 5%. From a practical point of view density-based solvers are not

suitable for flows with Mach number below that limit [182]

Even though all fluids are compressible in an absolute sense, the incompressible flow approximation can be assumed when the flow speed is small enough compared with the speed of sound of the medium. At this speed, the resolution of the compressible Navier-Stokes with an all-speed scheme could not be practical, motivated by the small time step required due the large disparity between the acoustic and the flow speed. In Chapter 4, the incompressible Navier-Stokes equations will be analyzed.

Chapter 4

A high-order FV-MLS formulation for incompressible flows.

4.1 Introduction

Even though all fluids are compressible in an absolute sense, the incompressible flow approximation can be assumed when the flow speed is small enough compared with the speed of sound of the medium. Many authors have set that limit at $M < 0.3$ [59, 182, 220]. From a practical point of view, density-based solvers require small time steps to be used in flows where the Mach number is low in all the computational domain [182], due the stiffness of the equations at the incompressibility limit [59]. The reason is that the time step is determined by the larger of the two velocities that the information travels: the velocity of the flow and the speed of sound. Therefore, in the low speed limit, the required time step may be much smaller that the time step required for an incompressible solver.

Numerical solution of the incompressible Navier-Stokes equations has a great interest due to its wide range of applications, such as low speed aerodynamics, biomedical fluid flow and hydrodynamics. Concerning turbomachines, there is interest on wind, tidal and hydraulic turbines in power-generation, due to the low operational Mach numbers, where the incompressible assumption can be applied.

The main problem with numerical solutions of incompressible flow is the difficulty when coupling changes of the velocity field with changes in the pressure field while satisfying the continuity equation. The reason is the absence of a transient term in the continuity equation, so the equations are decoupled and the continuity equation can be considered as constraint that the velocity field has to satisfy.

Pressure based methods, the artificial compressibility method and methods based

on derived variables are the three main approaches for solving the incompressible Navier-Stokes equations. The Marker And Cell (MAC) method [72] was one of the first methods for the numerical solution of incompressible flow using a derived Poisson equation for the pressure in order to satisfy mass conservation. In this approach, the pressure is used as a mapping parameter to satisfy the continuity equation. The method of artificial compressibility proposed by Chorin [29] introduces a pseudo-time derivative of the density into the continuity equation. The artificial compressibility method introduces a delay between the flow disturbance and its effect on the pressure field, whereas in a true incompressible flow, the pressure field is affected instantaneously by a disturbance. The major drawback of this method is the need for the definition of an artificial compressibility parameter, which is specific for each problem. The original form of Chorin's method was developed for steady state problems, and Peyret and Taylor [164] extended it to a time accurate formulation. Later, the method was fully extended to general three dimensions by Kwak et al. [108].

In order to remove the pressure from the formulation, different approaches introducing other variables instead have been developed. The most common approach of this kind is the stream function vorticity method [109]. The extension of this approach to 3D problems adds much complexity to the formulation and is more expensive that methods which solve velocity and pressure [108].

There are usually two kinds of grid arrangements used to solve the incompressible Navier-Stokes equations on a finite volume framework: staggered grids and non-staggered (collocated) grids. On the staggered grids, variables are stored at different locations, shifting the half of a control volume in each coordinate direction, as shown in figure 4.1.

The main advantage of the staggered grid approach is that no interpolation is required, since the variables are stored where they are needed. However, this arrangement presents difficulties in the application to unstructured and curvilinear grids. This difficulty increases when one deals with 3D problems. On the other hand, in collocated grids, vector variables and scalar variables are stored at the same locations, usually at the centroid of the control volume. This approach presents the possibility of checkerboard oscillations due to the central-difference discretization of the pressure. In order to circumvent the checkerboard oscillations, Rhie and Chow [174] proposed the Original Momentum Interpolation Method (OMIM), which interpolates the velocity in a manner that the velocity and pressure decoupling is removed. The Semi Implicit Method for Pressure Linked Equations, better known as the SIMPLE algorithm, is a pressure correction method. It uses the pressure as a mapping parameter in order to satisfy the decoupled equations. The main advantage of this approach is that the Poisson equation is not solved, since it is approximated with a pressure correction equation. Comparison between the staggered grid and collocated grid [27, 28]

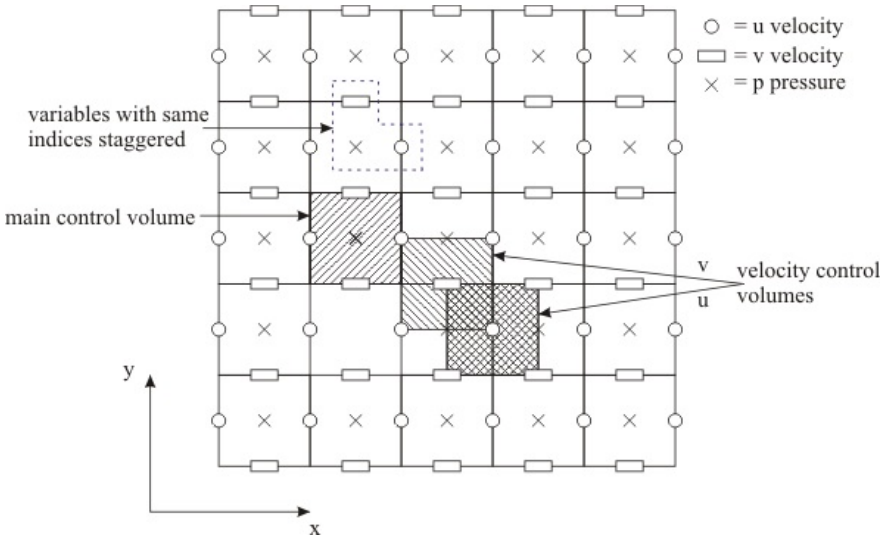


Figure 4.1: Staggered grid variables arragement. Figure taken from [85]

showed that the SIMPLE-like algorithms on collocated grids can provide similarly accurate results and convergence rates as those on staggered grid [23]. The SIMPLE method and its versions are currently widely used to solve the incompressible Navier-Stokes equations. Further details of the method and its variants can be found in [160].

In this Chapter a novel high-order numerical method for the resolution of the incompressible Navier-Stokes equations is presented. The great geometrical flexibility offered by unstructured grids makes them highly effective for dealing with complex geometries, that are more typical of practical engineering applications. Therefore, the new scheme is based on a collocated grid arrangement, and the pressure-based SIMPLE algorithm is used to solve the equations.

This Chapter can be summarized as follows: First, the governing equations and the numerical discretization are shown in sections 4.2 and 4.3. Finally, numerical results are presented in Section 4.4 to show the accuracy and robustness of the proposed method by solving different steady and unsteady benchmark problems in two dimensions.

4.2 Governing equations

The two-dimensional Navier-Stokes equations in Cartesian coordinates can be written in the following non dimensional form as

$$\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{U} \cdot (\nabla \boldsymbol{U}) = -\nabla p + \frac{1}{Re}(\Delta \boldsymbol{U}) \tag{4.1}$$

$$\nabla \cdot \boldsymbol{U} = 0 \tag{4.2}$$

where $\boldsymbol{U} = (u, v)^T$ is the velocity field, $p(x, y, t)$ is the pressure variable and Re denotes the Reynolds number. The Reynolds number is a dimensionless number that gives a measure of the ratio of inertial forces. It is defined like:

$$Re = \frac{\rho u_\infty D}{\mu} \tag{4.3}$$

where ρ is the density of the fluid, u_∞ is the reference velocity and D is the characteristic length.

The system of equations (4.1) and (4.2) is decoupled and the continuity equation acts as a constraint in the velocity. Since the equations are decoupled, the primary concern is how to obtain a flow field that satisfies both equations. A major obstacle to the numerical solution is to enforce the incompressility requirement. Since there is no time evolution term in the continuity equation, the time marching schemes developed for compressible flow solvers can not be applied. A quick overview of the different approaches is discussed below. The methods can be classified in three major groups:

- Pressure based methods
- Artifical Compressibility Method
- Methods Based on Derived Variables

4.2.1 Pressure based methods

The compressible flow has been usually computed using a density based solver. Since the density is constant for an incompressible flow, a different approach needs to be taken. Thus, the so-called pressure based formulation is used to keep the pressure field from oscillating. In order to couple the equations, this method first solves the pressure field, so that a divergence free velocity field is maintained at every time step. In 1965, Harlow and Welch published the method Marked And Cell (MAC) [29], which was the first method for incompressible flow using a derived Poisson equation for the pressure in order to conserve mass. In this approach, the pressure is used as a mapping parameter to satisfy the continuity equation.

The basic procedure for these methods is as follows:

1. Computation of a velocity field from the solution of the momentum equations with a guessed pressure field.

2. Computation of the pressure field from the Poisson equation with the previously computed velocity field.

3. Correct the velocity field with the new pressure field in order to satisfy the continuity equation.

4. Repeat steps 1 to 3 until the momentum and the continuity equations are satisfied.

The major drawback of the MAC method is that a Poisson equation must be solved for pressure each time step. A direct solver is only available for simple 2D cases, but for most of the 2D and 3D cases an iterative solver is the best available choice. However it will significantly slow down the overall computational efficiency.

In a steady state problem there is a need to compute the Poisson equation several times and the exact pressure it is only needed when the solution has converged. The SIMPLE algorithm [20] and its variants, SIMPLE-Revised (SIMPLER) [160], SIMPLE-Corrected (SIMPLEC) [211], SIMPLE-Corrected (SIMPLEC) [211] and Pressure Implicit with Splitting Operators (PISO) [87], approximate the solution to the Poisson equation. This methods were originally designed for steady state solutions, but they have been extended to unsteady flows [59, 220].

4.2.2 Artificial Compressibility Method

Large advances in CFD have been made in conjunction with the field of aerodynamics. The possibility of using some of the compressible flow algorithms could be of significant interest. In order to do this, the artificial compressibility method proposed by Chorin [29] introduces a pseudo-time derivative of the density into the continuity equation.

In equation (4.4), the artificial compressibility continuity equation is shown.

$$\frac{\partial \rho}{\partial t^*} + \nabla \cdot \mathbf{U} = 0 \tag{4.4}$$

Given $\beta^2 = \frac{\partial p}{\partial \rho}$, where β is the compressibility parameter and it represents the speed of the artificial pressure waves and t^* is a pseudo-time, the continuity equation can be written as

$$\frac{1}{\beta}\frac{\partial p}{\partial t^*} + \nabla \cdot \mathbf{U} = 0 \tag{4.5}$$

Physically, this means that waves of finite speed are introduced into the incompressible flow field as a medium to distribute the pressure. Numerically the pseudo term changes the character of the continuity equation from elliptic to hyperbolic. This allows the system of equations to be solved with a variety of time marching schemes developed

for compressible flow solvers. In a true incompressible flow, the pressure field is affected instantaneously by a disturbance in the flow, but with the artificial compressibility method, there will be a delay between the flow disturbance and its effect on the pressure field. The major drawback of this method is that the value of β needs to be found for a given problem.

The original form of this method was developed for steady state problems [29]. Peyret [164] was the first to extended it to a time accurate formulation. The method was later fully extended to general three dimensions by Kwak in 1984 [108].

4.2.3 Methods Based on Derived Variables

To avoid solving the pressure directly, other approaches that have been developed introduce other variables, allowing elimination of pressure from the formulation. The most common is the stream function vorticity method [109]. This approach has been used since the early days of CFD to solve two dimensional problems.

The extension of this approach to three dimensions adds much complexity to the formulation and it is more expensive that methods which solve velocity and pressure.

4.3 Numerical discretization of the incompressible Navier-Stokes equations

The numerical method presented in this section is based on the Semi Implicit Method for Pressure Linked Equations method, better known as the SIMPLE algorithm. This algorithm was first proposed by Caretto in 1972 [20], and later Patankar and Spalding published the algorithm as it is known today [159].

In order to solve the system formed with equations (4.1) and (4.2), the SIMPLE algorithm [160] is used. This method is englobed in the pressure correction method, so it uses the pressure as a mapping parameter in order to satisfy the decoupled equations. A schematic flowchart of the unsteady SIMPLE algorithm is shown in Figure 4.2. That is, for each time step, a number of inner iterations are needed until the velocity field $\boldsymbol{U} = (u, v)^T$ satisfies equations (4.1) and (4.2). For every inner iteration, the velocity field is predicted based on the momentum equation (4.1). Next, the velocity and pressure are corrected in order to satisfy the continuity equation (4.2). The corrections are obtained from a pressure correction equation derived from (4.1) and (4.2).

The main advantage of this method is that the Poisson equation is not solved. Instead, it is approximated with a pressure correction equation. The SIMPLE method and its versions are widely used to solve the incompressible Navier-Stokes equations. Further

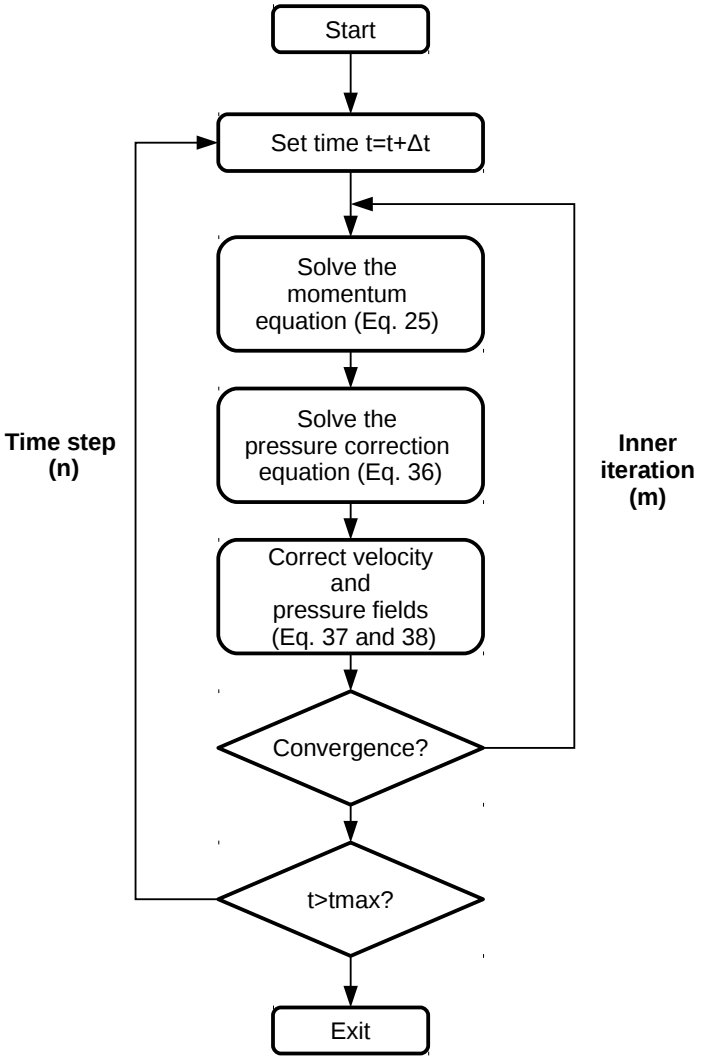


Figure 4.2: The unsteady SIMPLE algorithm flowchart

details of the SIMPLE, SIMPLER, SIMPLEC and PISO can be found in [160].

4.3.1 Momentum equation

The basic finite volume discretization stems from the integral form over the control volume Ω_I of equation (4.1).

$$\int_{\Omega_I} \frac{\partial \boldsymbol{U}}{\partial t} d\Omega + \int_{\Omega_I} \boldsymbol{U} \cdot (\nabla \boldsymbol{U}) d\Omega = -\int_{\Omega_I} \nabla p d\Omega + \frac{1}{Re} \left(\int_{\Omega_I} (\Delta \boldsymbol{U}) d\Omega \right) \tag{4.6}$$

Applying the divergence theorem for the convective, pressure and diffusive terms of equation (4.6), the following equation is obtained:

$$\int_{\Omega_I} \frac{\partial \boldsymbol{U}}{\partial t} d\Omega + \int_{\Gamma_I} \boldsymbol{U} \cdot (\boldsymbol{U} \cdot \boldsymbol{n}) d\Gamma = -\int_{\Gamma_I} p \cdot \boldsymbol{n} d\Gamma + \frac{1}{Re} \left(\int_{\Gamma_I} (\nabla \boldsymbol{U} \cdot \boldsymbol{n}) d\Gamma \right) \tag{4.7}$$

where Γ_I is the control volume perimeter and $\boldsymbol{n} = (n_x, n_y)^T$ is the unitary exterior normal of the contour.

Temporal discretization

The temporal integral is solved as

$$\int_{\Omega_I} \frac{\partial \boldsymbol{U}}{\partial t} d\Omega = V_I \frac{\partial \boldsymbol{U}_I}{\partial t} \tag{4.8}$$

where V_I is the volume of the control volume and $\boldsymbol{U}_I$ is the average value of the variables $\boldsymbol{U} = (u, v)^T$ at cell I. Using a second-order implicit discretization [59], given by

$$\frac{\partial \boldsymbol{U}_I}{\partial t} = \frac{3\boldsymbol{U}_I^{n+1} - 4\boldsymbol{U}_I^n + \boldsymbol{U}_I^{n-1}}{2\Delta t} \tag{4.9}$$

where n indicates the time step. As seen in Figure 4.2, for each time step an iterative process is performed. Therefore, equation (4.9) is approximated as

$$\frac{\partial \boldsymbol{U}_I}{\partial t} \approx \frac{3\boldsymbol{U}_I^{m+1,n+1} - 4\boldsymbol{U}_I^n + \boldsymbol{U}_I^{n-1}}{2\Delta t} \tag{4.10}$$

In equation (4.10), the superscript $m+1, n+1$ indicates the $m+1$ inner iteration level at time step $n+1$. At inner convergence, $\boldsymbol{U}_I^{n+1} = \boldsymbol{U}_I^{m+1,n+1}$ and equation (4.9) is satisfied.

Diffusive terms

Applying Gauss quadrature to evaluate the integral of the diffusive term in equation (4.7), leads to

$$\frac{1}{Re}\left(\int_{\Gamma_I}(\nabla \boldsymbol{U}\cdot\boldsymbol{n})d\Gamma\right)=\frac{1}{Re}\sum_{j=1}^{N_f}\sum_{ig=1}^{N_G}\left[\nabla \boldsymbol{U}_j\cdot\hat{\boldsymbol{n}}_j\right]_{ig}\mathcal{W}_{ig} \tag{4.11}$$

where N_f is the number of faces of the control volume, N_G represents the number of quadrature points, $\mathcal{W}_{ig}$ is the corresponding quadrature weight for the quadrature point at cell faces, denoted with subscript ig, and $\hat{\boldsymbol{n}}_j$ is the unitary normal $\boldsymbol{n}$ times the area of the face j. The gradient $\nabla \boldsymbol{U}_j$ is computed directly at integration points using MLS derivatives as

$$\nabla \boldsymbol{U}_j=\sum_{l=1}^{n_{\boldsymbol{x}}}\nabla N_l^g(\boldsymbol{x}_j)\boldsymbol{U}_l \tag{4.12}$$

Note that the neighbors $n_{\boldsymbol{x}}$ are the centroids of neighboring cells.

As explained in Chapter 2, the non-interpolant nature of MLS is important for the imposition of boundary conditions. Dirichlet boundary conditions for equation (4.11) are imposed in equation (4.12) through the ghost cells. Thus, for example, the zero velocity at a wall edge is imposed at ghost cells placed on this edge. The value $\nabla \boldsymbol{U}_j$ is then computed with equation (4.12) using a stencil such as the one shown in Figure 2.3. For inlet boundary conditions, the same approach has been used. For outflow boundary conditions, equation (4.12) is not used and the value $\nabla \boldsymbol{U}_j\cdot\hat{\boldsymbol{n}}_j=(0,0)^T$ is directly imposed in equation (4.11).

Convective terms

The convective term of the incompressible Navier-Stokes equations is discretized as

$$\int_{\Gamma_I}\boldsymbol{U}\cdot(\boldsymbol{U}\cdot\boldsymbol{n})d\Gamma=\sum_{j=1}^{N_f}\sum_{ig=1}^{N_G}\left[H_j\boldsymbol{U}_j\right]_{ig}\mathcal{W}_{ig} \tag{4.13}$$

where H_j is defined as

$$H_j=\hat{\boldsymbol{U}}_j\cdot\hat{\boldsymbol{n}}_j \tag{4.14}$$

The value of $\hat{\boldsymbol{U}}_j$ is obtained with the Momentum Interpolation Method (MIM) [230], which will be explained in section 4.3.2.

In order to achieve higher-order accuracy, a deferred correction is used. The deferred correction uses a low-order implicit approximation improved each inner iteration with an explicit approximation of the error between the low-order and the higher-order discretizations. At inner convergence, only the higher-order discretization will remain. This approach was first reported by Khosla and Rubin [104], and Demirdžić at al. [44] proposed the use of this technique with the SIMPLE algorithm. As pointed in [44, 59], the deferred correction is used to increase the stability of the computational procedure while preserving the numerical accuracy.

Using this approach, the term $\boldsymbol{U}_j$ in equation (4.13) is discretized as

$$\boldsymbol{U}_j = \left(\boldsymbol{U}_j^{LO}\right)^{m+1,n+1} + \left(\boldsymbol{U}_j^{HO} - \boldsymbol{U}_j^{LO}\right)^{m,n+1} \tag{4.15}$$

where the superscript $m, n+1$ stands for the previous inner iteration at the current time step $n+1$. In equation (4.15), the higher-order convective term, $\boldsymbol{U}_j^{HO}$, is treated explicitly from the previous inner iteration (m).

In order to achieve arbitrary higher-order we compute the term $\boldsymbol{U}_j^{HO}$ directly at integration points with MLS as

$$\boldsymbol{U}_j^{HO} = \sum_{k=1}^{n_{\boldsymbol{x}}} N_k^g(\boldsymbol{x}_j)\boldsymbol{U}_k \tag{4.16}$$

The term $\boldsymbol{U}_j^{LO}$ is computed using a first order upwind, defined as

$$\boldsymbol{U}_j^{LO} = \begin{cases} \boldsymbol{U}_I & , H_j \geq 0 \\ \\ \boldsymbol{U}_N & , H_j < 0 \end{cases} \tag{4.17}$$

where the subscript N stands for the cell that shares the face j with cell I, as seen in Figure 4.3

Equation (4.17) can be equivalently expressed as

$$\boldsymbol{U}_j^{LO} = \boldsymbol{U}_I \, \max(H_j, 0) + \boldsymbol{U}_N \, \min(H_j, 0) \tag{4.18}$$

On a boundary edge, Dirichlet boundary conditions for equation (4.13) are imposed by setting the prescribed known value $\boldsymbol{U}_j$. For example, for a wall edge $\boldsymbol{U}_j = (0,0)^T$.

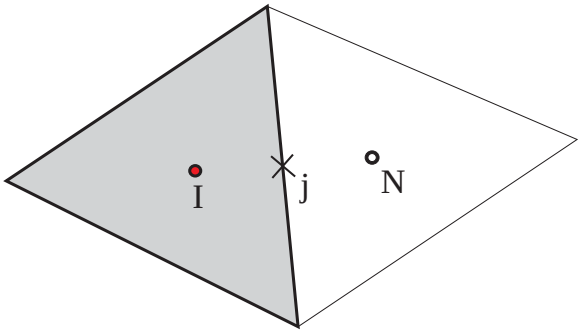


Figure 4.3: Schematic representation of cell I and its integration point j.

Pressure terms

The pressure term of the momentum equation is computed as

$$\int_{\Gamma_I} p \cdot \boldsymbol{n} d\Gamma = \sum_{j=1}^{N_f} \sum_{ig=1}^{N_G} \left[p_j \cdot \hat{\boldsymbol{n}}_j \right]_{ig} \mathcal{W}_{ig} \tag{4.19}$$

where the values of the pressure at integration points p_j are approximated directly by MLS from the values of the pressure at the neighboring centroids (p_k), as indicated in equation (4.20).

$$p_j = \sum_{k=1}^{n_{\boldsymbol{x}}} N_k^g(\boldsymbol{x}_j) p_k \tag{4.20}$$

This discretization gives a higher-order and very accurate approximation of the values of the pressure at integration points.

On a boundary edge, the Neumann boundary condition $\partial p / \partial n = 0$ has to be imposed. In this work we have used an approach based on the work of Kampanis [96]. The idea is to compute the value of the pressure at the ghost cells in such a way that the condition $(\nabla p)_j \cdot \hat{\boldsymbol{n}}_j = 0$ is verified. That is

$$(\nabla p)_j \cdot \hat{\boldsymbol{n}}_j = \sum_{l=1}^{n_{\boldsymbol{x}}} \nabla N_l^g(\boldsymbol{x}_j) p_l \cdot \hat{\boldsymbol{n}}_j = 0 \tag{4.21}$$

Discretized momentum equation

The application of equations (4.8), (4.11), (4.13) and (4.19) to equation (4.7) leads to the following expression

$$\begin{aligned} & V_I \frac{3\boldsymbol{U}_I^{m+1,n+1} - 4\boldsymbol{U}_I^n + \boldsymbol{U}_I^{n-1}}{2\Delta t} \\ + & \sum_{j=1}^{N_f}\sum_{ig=1}^{N_G} \left[H_j^{m,n+1} \left(\left(\boldsymbol{U}_j^{LO}\right)^{m+1,n+1} + \left(\boldsymbol{U}_j^{HO} - \boldsymbol{U}_j^{LO}\right)^{m,n+1} \right) \right]_{ig} \mathcal{W}_{ig} \\ = & -\sum_{j=1}^{N_f}\sum_{ig=1}^{N_G} \left[p_j^{m,n+1} \cdot \hat{\boldsymbol{n}}_j \right]_{ig} \mathcal{W}_{ig} + \frac{1}{Re} \sum_{j=1}^{N_f}\sum_{ig=1}^{N_G} \left[\nabla \boldsymbol{U}_j^{m+1,n+1} \cdot \hat{\boldsymbol{n}}_j \right]_{ig} \mathcal{W}_{ig} \end{aligned} \tag{4.22}$$

Since a Semi-Implicit formulation is used, for each time step a number of inner iterations are needed until equations (4.1) and (4.2) are satisfied. The superscript $m+1$ refers to the current inner iteration, m to the previous inner iteration, $n+1$ refers to the current time iteration, while the previous time iterations are denoted by index n and $n-1$.

4.3.2 Velocity and pressure correction

Once the velocity field is computed from equation (4.22), it is required to impose the continuity constraint given by equation (4.2). In order to satisfy this constraint, a pressure correction algorithm is used. First, the continuity equation (4.2) is discretized as

$$\int_{\Omega_I} \nabla \cdot \boldsymbol{U} d\Omega = \int_{\Gamma_I} \boldsymbol{U} \cdot \boldsymbol{n} d\Gamma = \sum_{j=1}^{N_f}\sum_{ig=1}^{N_G} \left[\hat{\boldsymbol{U}}_j \cdot \hat{\boldsymbol{n}}_j \right]_{ig} \mathcal{W}_{ig} \tag{4.23}$$

where the value $\hat{\boldsymbol{U}}_j$ is approximated using the Momentum Interpolation Method (MIM) to avoid checkerboard problems, as it will be described in the following section. This value of the velocity does not satisfy the continuity constraint, and in order to impose it, the SIMPLE algorithm is used. In the SIMPLE algorithm, a pressure correction equation is obtained. As it is indicated in Figure 4.2, once the pressure correction is solved, the velocity and pressure field are corrected. This is shown in the following sections.

Momentum Interpolation Method

It is known that the numerical solution of the incompressible Navier-Stokes using a collocated variable arrangement produces checkerboard oscillations [230]. The oscillations are caused by the decoupling of pressure and velocity in the discretized equation due by the central difference discretization of the pressure. Rhie and Chow [174] reported that if

the value $\hat{\boldsymbol{U}}_j$ is interpolated as usual, it presents the so-called *odd-even* dependency of the pressure, causing the checkerboard oscillation.

The origin of the checkerboard oscillations will be briefly explained and different methods to overtake it will be reviewed.

In the following explanation a one-dimensional structured grid will be used with the notation as seen in Figure 4.4.

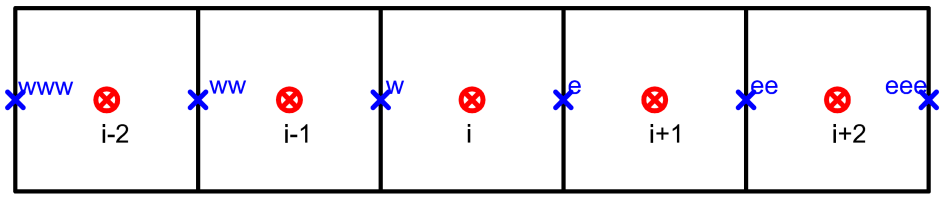


Figure 4.4: One-dimensional structured grid

In the u-momentum equation the pressure term is obtained as:

$$\int_{\Omega_I} \frac{\partial p}{\partial x} d\Omega = \sum_{j=1}^{N_f} \sum_{ig=1}^{N_G} \left[p_j \hat{n}_{xj} \right]_{ig} \mathcal{W}_{ig} \tag{4.24}$$

If one integration point is used, the equation can be written as

$$\int_{\Omega_I} \frac{\partial p}{\partial x} d\Omega = \left(p \hat{n}_x \right)_e + \left(p \hat{n}_x \right)_w \tag{4.25}$$

where p_e and p_w are the pressure interpolated at the right and left integration points. In Figure 4.4 a schematic representation is shown.

If $\Delta x = 1$, then $\left(\hat{n}_x\right)_e = -\left(\hat{n}_x\right)_e = 1$. The pressure discretization reads

$$\int_{\Omega_I} \frac{\partial p}{\partial x} d\Omega = p_e - p_w \tag{4.26}$$

For simplicity, the values p_e and p_w are linearly interpolated, $p_e = 0.5(p_{i+1} + p_i)$ and $p_w = 0.5(p_{i-1} + p_i)$. Introducing these values in equation (4.26),

$$\int_{\Omega_I} \frac{\partial p}{\partial x} d\Omega = p_e - p_w = 0.5(p_{i+1} + p_i) - 0.5(p_{i-1} + p_i) = 0.5(p_{i+1} - p_{i-1}) \tag{4.27}$$

In the pressure term of the u-momentum equation for cell i, equation (4.27), the pressure p_i is not considered. Therefore the momentum equations only produce links between pressures at alternated nodes if a collocated grid arrangement is used. Thus, a field

of the form A-B-A-B-A-B will be constant for this discretization. For example, in Figure 4.5, the checkerboarded pressure field will be taken as constant because there are only links between pressures at alternated nodes. Now, focusing in the discretized continuity equation

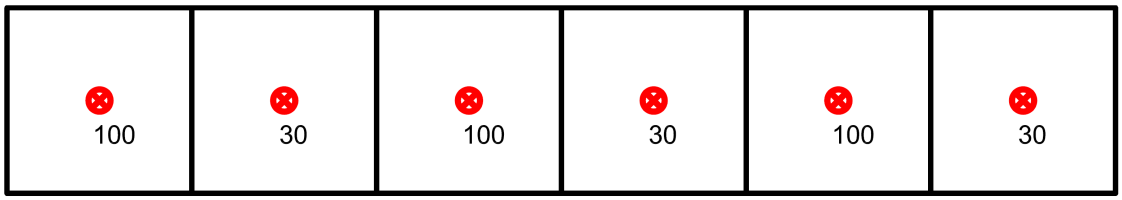


Figure 4.5: Checkerboard pressure field

$$\sum_{j=1}^{N_f} u_j \hat{n}_{xj} = u_e - u_w \tag{4.28}$$

where the values at cell faces can be interpolated as $u_e = 0.5(u_i + u_{i+1})$ and $u_w = 0.5(u_i + u_{i-1})$. The continuity equation can be rewritten

$$\sum_{j=1}^{N_f} u_j \hat{n}_{xj} = u_e - u_w = 0.5(u_i + u_{i+1}) - 0.5(u_i + u_{i-1}) = 0.5u_{i+1} - 0.5u_{i-1} \tag{4.29}$$

where the values u_{i+1} and u_{i-1} are computed with the discretized momentum equation.

In the computation of the divergence in cell i, neither the velocity nor the pressure fields of cell i appear in the continuity equation (4.29). Thus, both mass and momentum equations lead to *odd-even* decoupling. Using the previous interpolation of the continuity equation (4.29), an oscillating velocity and pressure fields can be obtained. For example, in Figure 4.6 a one-dimensional pressure field is represented. Th use of the standard interpolation methods will lead to a decoupling of odd nodal values p_1, p_3, p_5... from even nodal values p_2, p_4, p_6....

There are two common remedies to avoid the checkerboard:

- A staggered grid arrangement.
- A collocated grid arrangement with different interpolation approaches, such as the Momentum Interpolation Method (MIM).

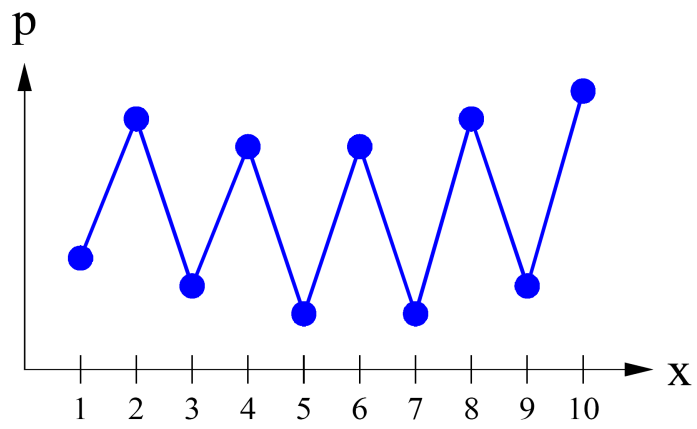


Figure 4.6: One dimensional checkerboarded pressure field

This weakness was circumvented by Rhie and Chow [174], when they proposed a momentum-based interpolation method to interpolate mass fluxes on cell faces. This method, the Original Momentum Interpolation Method (OMIM), interpolates the velocity in a manner that the velocity and pressure decoupling is removed. In their approach, they mimic a staggered-grid discretization by expressing the discrete mass conservation equation in terms of the discrete mass fluxes across cell faces. The Rhie and Chow approach has been widely used for decades [86, 199, 229]. The basic idea of the Momentum Interpolation Method (MIM) is to remove the interpolated pressure gradient term that comes from the momentum equation and then add the pressure gradient calculated directly at cell face. Majumdar [129] and Miller [134] reported that solutions of steady-state problems from Rhie and Chow's original MIM are dependent on the underrelaxation factor. In order to eliminate this dependency, Majumdar proposed an iterative algorithm. In 1999, Choi [28] reported that the solution using the original MIM scheme is time-step size-dependent, and proposed a modified Momentum Interpolation Method. In 2002, Yu et al. [230] observed that the solutions obtained with Choi's scheme are still time-step size-dependent, though the dependence is quite small. Yu et al. [230] proposed a different interpolation technique and they demonstrated numerically and mathematically that the resulting scheme was time-step size and underrelaxation factor independent. The original formulation has been recently improved to solve these problems [233].

Following the idea of the OMIM, the interpolated value $\hat{\boldsymbol{U}}_j$ is computed removing the pressure gradient term that comes from the momentum equations (term $\left(\overline{\nabla p_I}\right)_j$ in equation (4.30)) and adding the pressure gradient computed directly at integration point

(term $(\nabla p)_j$ in equation (4.30)). The reader is referred to [174] for more details. The OMIM uses the following correction to the velocity to prevent checkerboard oscillations

$$\hat{\boldsymbol{U}}_j = \boldsymbol{U}_j^* + \left(\frac{V_I}{a_I}\right)_j \left[\left(\overline{\nabla p_I}\right)_j - \nabla p_j\right] \tag{4.30}$$

where $\boldsymbol{U}_j^*$ is the solution obtained with the momentum equations (4.22), $\boldsymbol{U}^*$, approximated at integration points, and a_I denotes the diagonal term of the mass matrix of the momentum equations for cell I, that is

$$a_I = \frac{3V_I}{2\Delta t} + \sum_{j=1}^{N_f}\sum_{ig=1}^{N_G}\left[\max(H_j,0) - \frac{1}{Re}(\nabla N_I(\boldsymbol{x}_j)\cdot\hat{\boldsymbol{n}}_j)\right]_{ig}\mathcal{W}_{ig} \tag{4.31}$$

Note that all terms with the subscript j of equation (4.30) need to be approximated since the known values are located at the centroid of the cells. Typical approaches based on the Rhie and Chow interpolation use a linear interpolation to obtain the values of $\boldsymbol{U}^*$, $\left(\frac{V_I}{a_I}\right)$, $\left(\overline{\nabla p_I}\right)_j$ and ∇p_j at the integration point j [129, 174, 230]. These values are obtained going beyond the linear interpolation by using higher-order MLS approximations. That is

$$\boldsymbol{U}_j^* = \sum_{k=1}^{n_{\boldsymbol{x}}} N_k^g(\boldsymbol{x}_j)\boldsymbol{U}_k^* \tag{4.32}$$

$$\left(\frac{V_I}{a_I}\right)_j = \sum_{k=1}^{n_{\boldsymbol{x}}} N_k^g(\boldsymbol{x}_j)\left(\frac{V_I}{a_I}\right)_k \tag{4.33}$$

$$\left(\overline{\nabla p_I}\right)_j = \sum_{k=1}^{n_{\boldsymbol{x}}} N_k^g(\boldsymbol{x}_j)\nabla p_k \tag{4.34}$$

In equation (4.34), the pressure gradient (∇p_k) is obtained at the neighboring cell centroid k as

$$\nabla p_k = \sum_{l=1}^{n_{\boldsymbol{x}}} \nabla N_l(\boldsymbol{x}_k)p_l \tag{4.35}$$

Following the idea of Rhie and Chow, the value ∇p_j of equation (4.30) is computed directly at the integration point j using MLS

$$\nabla p_j = \sum_{l=1}^{n_{\boldsymbol{x}}} \nabla N_l^g(\boldsymbol{x}_j) p_l \tag{4.36}$$

Notice that ∇p_j and $\left(\overline{\nabla p_I}\right)_j$ are different. The value ∇p_j is computed in (4.36) directly at the integration point j using the derivatives of the MLS shape functions and the values of the pressure at the neighboring cells. The value $\left(\overline{\nabla p_I}\right)_j$ is computed by approximating the pressure gradient at the integration point j using MLS shape functions and the values ∇p computed at the $n_{\boldsymbol{x}}$ neighboring cells centroids.

It is known that the Rhie-Chow interpolation is time dependent. The modification proposed in [233] removes the time dependency for first order time integration schemes. Here, this formulation is extended for the second-order time integration used in this work.

$$\begin{aligned} \hat{\boldsymbol{U}}_j = \boldsymbol{U}_j^* \;&+\; \left(\frac{V_I}{a_I}\right)_j \left[\left(\overline{\nabla p_I}\right)_j - \nabla p_j\right] \\ &+\; \frac{4}{2\Delta t}\left(\frac{V_I}{a_I}\right)_j \left\{\left(\frac{V_I}{a_I}\right)_j \left[\left(\overline{\nabla p_I}\right)_j - \nabla p_j\right]\right\}^{n} \\ &-\; \frac{1}{2\Delta t}\left(\frac{V_I}{a_I}\right)_j \left\{\left(\frac{V_I}{a_I}\right)_j \left[\left(\overline{\nabla p_I}\right)_j - \nabla p_j\right]\right\}^{n-1} \end{aligned} \tag{4.37}$$

where the superscript n denotes the temporal iteration. The value of $\hat{\boldsymbol{U}}_j$ given by equation (4.37) is used in equations (4.14) and (4.41).

Notice that all the interpolated values are obtained employing a higher-order MLS approximation instead of the usual linear interpolation.

On boundary edges, Dirichlet boundary conditions are imposed by setting the prescribed values of $\hat{\boldsymbol{U}}_j$ at equation (4.37).

Pressure Correction Equation

The finite volume discretization of the continuity equation was presented in equation (4.23). For the sake of clarity, it will be rewritten in equation (4.38)

$$\int_{\Omega_I} \nabla \cdot \boldsymbol{U} d\Omega = \int_{\Gamma_I} (\boldsymbol{U} \cdot \boldsymbol{n}) d\Gamma = \sum_{j=1}^{N_f} \sum_{ig=1}^{N_G} \left[\hat{\boldsymbol{U}}_j \cdot \hat{\boldsymbol{n}}_j\right]_{ig} \mathcal{W}_{ig} \tag{4.38}$$

where the velocity $\hat{\boldsymbol{U}}_j$ is computed using equation (4.37). Since the system of equations formed with equations (4.1) and (4.2) is decoupled, the velocity obtained with equation (4.37) does not satisfy equation (4.2), that is

$$\int_{\Omega_I} \nabla \cdot \boldsymbol{U} d\Omega = \int_{\Gamma_I} \boldsymbol{U} \cdot \boldsymbol{n} d\Gamma = \sum_{j=1}^{N_f} \sum_{ig=1}^{N_G} \left[\hat{\boldsymbol{U}}_j \cdot \hat{\boldsymbol{n}}_j \right]_{ig} \mathcal{W}_{ig} \neq 0 \tag{4.39}$$

In order to satisfy the continuity equation the velocity and pressure are corrected each inner iteration. In the SIMPLE algorithm, the face velocity correction, $\boldsymbol{U}_j^{'}$, is defined as

$$\boldsymbol{U}_j^{'} = -\left(\frac{V_I}{a_I}\right)_j \left(\nabla p^{'}\right)_j \tag{4.40}$$

where $p^{'}$ is the pressure correction. Introducing the velocity correction equation (4.40) in the discretized continuity equation (4.23), the pressure correction equation is

$$\sum_{j=1}^{N_f} \sum_{ig=1}^{N_G} \left[\hat{\boldsymbol{U}}_j \cdot \hat{\boldsymbol{n}}_j \right]_{ig} \mathcal{W}_{ig} - \sum_{j=1}^{N_f} \sum_{ig=1}^{N_G} \left[\left(\frac{V_I}{a_I}\right)_j \left(\nabla p^{'}\right)_j \cdot \hat{\boldsymbol{n}}_j \right]_{ig} \mathcal{W}_{ig} = 0 \tag{4.41}$$

where $\left(\frac{V_I}{a_I}\right)_j$ and $\left(\nabla p^{'}\right)_j$ are obtained using MLS approximations and the value $\hat{\boldsymbol{U}}_j$ is computed as defined in equation (4.37). Note that in equation (4.41) the only unknown is $p^{'}$. This equation is solved using the Gauss-Seidel algorithm.

On boundary edges, Neumann boundary conditions for the pressure correction $p^{'}$ are imposed, that is, $\left(\nabla p^{'}\right)_j \cdot \hat{\boldsymbol{n}}_j = 0$. The prescribed value is directly imposed at equation (4.41).

Finally, the velocity and pressure fields are updated by

$$\boldsymbol{U}^{m+1,n+1} = \boldsymbol{U}^{*} + \boldsymbol{U}^{'} = \boldsymbol{U}^{*} - \frac{V_I}{a_I} \left(\nabla p^{'}\right)_I \tag{4.42}$$

$$p^{m+1,n+1} = p^{m,n+1} + \left(p^{'}\right)^{m+1,n+1} \tag{4.43}$$

where $\boldsymbol{U}^{*}$ is the velocity field obtained from the momentum equation (4.22).

The velocity $\hat{\boldsymbol{U}}_j$ obtained in equation (4.37) is only used in equations (4.14) and (4.41) to avoid checkerboard oscillations.

In summary, the resolution process is iteratively performed until velocity and pressure satisfy equations (4.1) and (4.2). At this point, inner convergence is reached and the

velocity and pressure fields are updated as $\boldsymbol{U}^{n+1} = \boldsymbol{U}^{m+1,n+1}$ and $p^{n+1} = p^{m+1,n+1}$. For clarity, the unsteady SIMPLE algorithm is briefly recalled:

1. Guess the pressure field p. For example, the pressure from previous inner iteration $p^{m,n+1}$.

2. Solve the momentum equation. Eq. (4.22).

3. Solve the pressure correction equation. Eq. (4.41).

4. Correct the cell centered velocities and the pressure field. Eq. (4.42), (4.43).

5. Check local convergence: when the L_2 error of the momentum and the continuity equation is less than a given tolerance the solution has inner convergence. If satisfied go to the next time step, otherwise, go to step 1 using the corrected values of pressure and velocity.

4.4 Numerical Results

This section presents numerical results for several test cases aimed at assessing the accuracy and efficiency of the proposed method for both steady and unsteady flow problems.

The first two examples are typical tests commonly used to evaluate the spatial order of convergence in steady and unsteady flows [54, 59, 64, 96, 110, 135, 162, 171, 188, 221].

The third problem is a benchmark problem widely used to validate the incompressible flow solver. The obtained results are compared with the reference solution obtained by Ghia [64] and with the results obtained with a commercial code, in order to evaluate the improvements of the new numerical method for the current state-of-art.

Finally, the proposed method is evaluated with the laminar flow around a cylinder.

4.4.1 Kovasznay Flow

In this first test the Kovasznay flow [107] is solved. As Fadel described [54], the Kovasznay flow is a good test for the incompressible Navier-Stokes formulation since this problem incorporates nonlinear effects. Given a domain $\Omega = [-0.5, 0.5] \times [-0.5, 0.5]$, the analytical solution is

$$
\begin{aligned}
u(x,y) &= 1 - e^{\alpha x}\cos(2\pi y) \\
v(x,y) &= \frac{\alpha}{2\pi} e^{\alpha x}\sin(2\pi y) \\
p(x,y) &= \frac{1}{2}\left(1 - e^{2\alpha x}\right)
\end{aligned}
\tag{4.44}
$$

where the parameter α is defined as

$$
\alpha = \frac{Re}{2} - \sqrt{\frac{Re^2}{4} + 4\pi^2}
\tag{4.45}
$$

The flow is initialized by enforcing the analytical solution given by the previous equations on the domain. Dirichlet boundary conditions are employed on all the boundaries where the values are prescribed with the analytical solution. The representation of the analytical condition is shown in Figure 4.7.

The Reynolds number employed is $Re = 40$ and different meshes of 32x32, 64x64, 128x128 and 256x256 are used in order to obtain the committed error an the spatial order of convergence. The formulation is tested with 1 and 2 integration points. The spatial order is computed as

$$
Order(\bullet) = \frac{log\left(\frac{L2_n(\bullet)}{L2_{2n}(\bullet)}\right)}{log2}
\tag{4.46}
$$

where $L2_n(\bullet)$ the L2 norm of the variable $(\bullet)$ on a $n \times n$ mesh and $L2_{2n}(\bullet)$ the L2 norm of the variable $(\bullet)$ on a $2n \times 2n$ mesh.

Tables 4.1and 4.2 show a convergence rate analysis for the proposed new scheme. The results obtained confirm that the formulation achieves the correct order of convergence for velocity and pressure when two Gauss points are employed with the cubic basis in the MLS formulation. When one Gauss point is employed, the order of convergence is stacked at two.

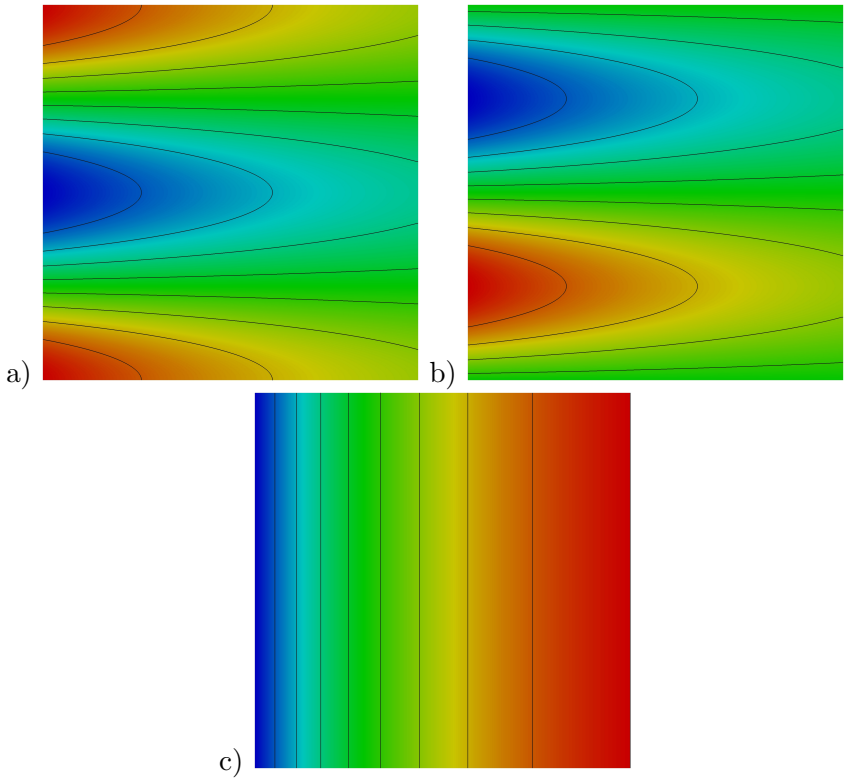


Figure 4.7: Analytical conditions of the Kovasznay flow for $Re = 40$. Figures a) and b) represents the u-velocity and v-velocity fields, and figure c) represents the pressure field and contours.

	u-velocity		v-velocity		pressure	
Mesh	L_2u	Order	L_2v	Order	L_2p	Order
32×32	5.4910^{-04}	-	2.5210^{-04}	-	5.2410^{-04}	-
64×64	1.2610^{-04}	2.12	5.9410^{-05}	2.08	1.1810^{-04}	2.15
128×128	3.0310^{-05}	2.06	1.4510^{-05}	2.03	2.7810^{-05}	2.09
256×256	7.4410^{-06}	2.03	3.6010^{-06}	2.01	6.7010^{-06}	2.05

Table 4.1: Accuracy orders of velocity components and pressure field for Kovasznay flow test case. Re=40, cubic MLS and one Gauss point is employed. The order of convergence is stacked at two.

	u-velocity		v-velocity		pressure	
Mesh	L_2u	Order	L_2v	Order	L_2p	Order
32×32	1.4110^{-05}	-	7.9910^{-06}	-	2.6410^{-05}	-
64×64	1.0010^{-06}	3.81	4.1510^{-07}	4.27	1.2010^{-06}	4.47
128×128	6.5510^{-08}	3.94	2.5110^{-08}	4.05	6.9610^{-08}	4.10
256×256	4.1710^{-09}	3.97	1.5510^{-09}	4.02	4.9810^{-09}	3.81

Table 4.2: Accuracy orders of velocity components and pressure field for Kovasznay flow test case. Re=40, cubic MLS and two Gauss points are employed. The expected order of convergence is obtained.

4.4.2 2D Taylor-Green Flow

In this section an unsteady example, namely the Taylor-Green vortex flow [110, 162, 171] is presented. It is an exact time-dependent solution to the incompressible Navier-Stokes equations in 2D. The flow decays in time at a rate controlled by viscosity. The analytical solution for the velocity and pressure fields corresponding to this flow in the domain $\Omega = [0, 2\pi] \times [0, 2\pi]$ is given by [110]

$$\begin{aligned} u(x,y) &= e^{\frac{-2t}{Re}} \cos(y) \sin(x) \\ v(x,y) &= -e^{\frac{-2t}{Re}} \cos(x) \sin(y) \\ p(x,y) &= \frac{e^{\frac{-4t}{Re}}}{4} (\cos(2x) + \cos(2y)) \end{aligned} \tag{4.47}$$

Here, the present FV-MLS formulation is employed using two Gauss points with the quadratic and cubic basis. The Reynolds number is set to 100. Tables 4.3, 4.4, 4.5 and 4.6 present the L_2 and L_∞ norms and corresponding orders of accuracy computed at $t = 10^{-2}$ with time steps of $\Delta t = 10^{-4}$. It is observed that the formal order of convergence is obtained in both third and fourth order MLS reconstructions. In Figure 4.9, the L_2 norm errors are plotted for the different mesh resolutions. The accuracy level for the pressure field is approximatively one order of magnitude lower than those given by the velocity field, fact that has been already reported in the literature by other authors [171].

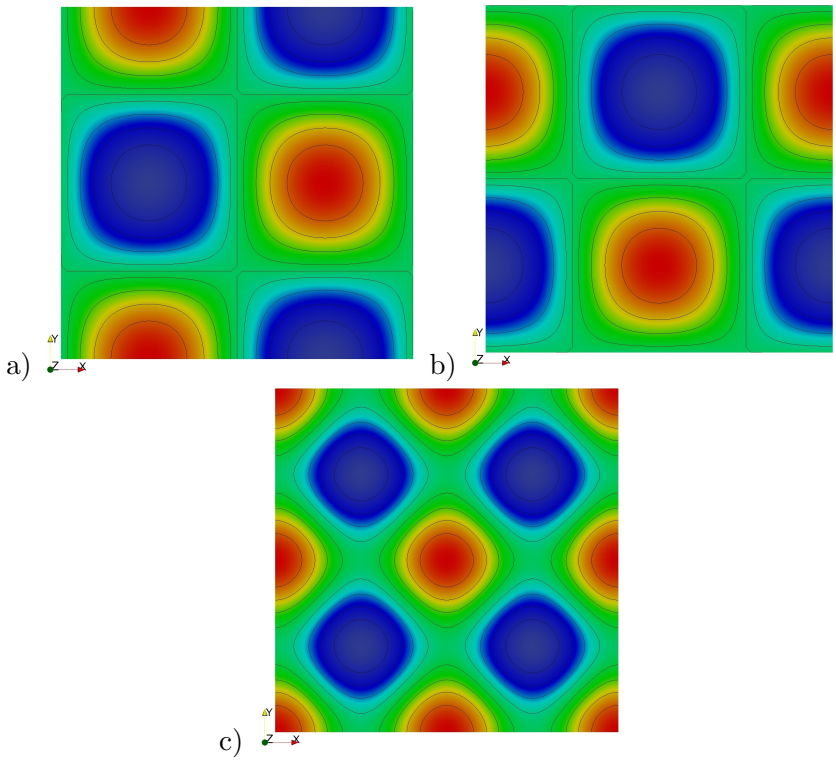


Figure 4.8: Initial conditions of the Taylor Green flow for $Re = 100$. Figures a) and b) represents the u-velocity and v-velocity fields, and figure c) represents the pressure field and contours.

	u-velocity		v-velocity		pressure	
Mesh	L_2u	Order	L_2v	Order	L_2p	Order
32×32	2.50×10^{-5}	-	2.50×10^{-5}	-	1.81×10^{-3}	-
64×64	2.57×10^{-6}	3.28	2.57×10^{-6}	3.29	1.22×10^{-4}	3.88
128×128	2.78×10^{-7}	3.21	2.79×10^{-7}	3.20	8.11×10^{-6}	3.92

Table 4.3: Accuracy orders and L_2 of velocity components and pressure field for the Taylor-Green vortex test case at $Re = 100$ and $t = 10^{-2}$ for the third-order FV-MLS scheme.

	u-velocity		v-velocity		pressure	
Mesh	$L_\infty u$	Order	$L_\infty v$	Order	$L_\infty p$	Order
32×32	2.91×10^{-4}	-	2.98×10^{-4}	-	2.72×10^{-2}	-
64×64	5.06×10^{-5}	2.55	5.05×10^{-5}	2.56	2.64×10^{-3}	3.36
128×128	7.26×10^{-6}	2.80	7.24×10^{-6}	2.80	2.10×10^{-4}	3.65

Table 4.4: Accuracy orders and L_∞ of velocity components and pressure field for the Taylor-Green vortex test case at $Re = 100$ and $t = 10^{-2}$ for the third-order FV-MLS scheme.

	u-velocity		v-velocity		pressure	
Mesh	$L_2 u$	Order	$L_2 v$	Order	$L_2 p$	Order
32×32	4.54×10^{-6}	-	4.56×10^{-6}	-	1.12×10^{-4}	-
64×64	2.76×10^{-7}	4.04	2.77×10^{-7}	4.04	5.48×10^{-6}	4.36
128×128	1.67×10^{-8}	4.04	1.70×10^{-8}	4.03	3.10×10^{-7}	4.14

Table 4.5: Accuracy orders and L_2 of velocity components and pressure field for the Taylor-Green vortex test case at $Re = 100$ and $t = 10^{-2}$ for the fourth-order FV-MLS scheme.

	u-velocity		v-velocity		pressure	
Mesh	$L_\infty u$	Order	$L_\infty v$	Order	$L_\infty p$	Order
32×32	9.53×10^{-6}	-	9.18×10^{-6}	-	4.71×10^{-4}	-
64×64	4.74×10^{-7}	4.33	4.73×10^{-7}	4.28	1.65×10^{-6}	4.84
128×128	3.54×10^{-8}	3.74	3.59×10^{-8}	3.72	7.90×10^{-7}	4.38

Table 4.6: Accuracy orders and L_∞ of velocity components and pressure field for the Taylor-Green vortex test case at $Re = 100$ and $t = 10^{-2}$ for the fourth-order FV-MLS scheme.

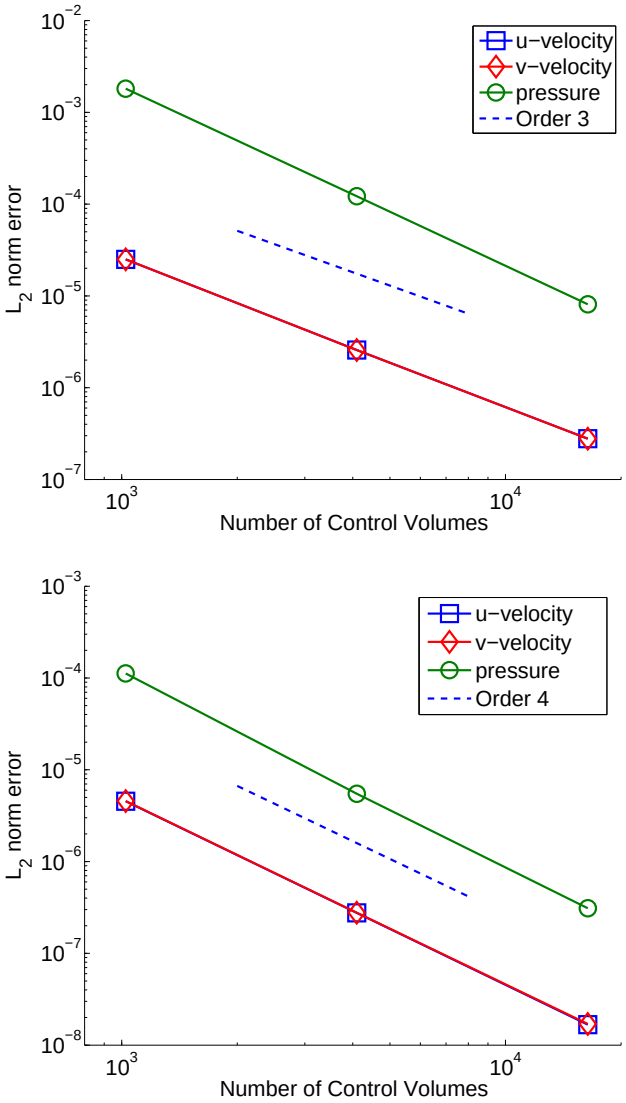


Figure 4.9: L_2 norm of error of velocity and pressure fields as function of the number of cells for the Taylor-Green vortex flow with $Re = 100$. The FV-MLS scheme computations are based on two Gauss points quadrature rules. Dashed line denotes the slope of the expected formal order of accuracy.

4.4.3 Lid-driven cavity Flow

The driven cavity flow problem is a classical benchmark problem used to validate an incompressible flow solver [54, 96, 64]. The square domain $\Omega = [0, 1] \times [0, 1]$ was discretized using an unstructured mesh to check the suitability of the method to deal with

irregular grids. The initial condition was zero velocity and zero pressure. The pressure on the boundaries is obtained by assuming zero normal pressure gradient. No-slip boundary conditions are imposed for all the edges except for the top boundary that represents a moving lid where the velocity field is defined by $u = 1$ and $v = 0$. Figure 4.10 shows the basic geometry of the problem and the boundary conditions.

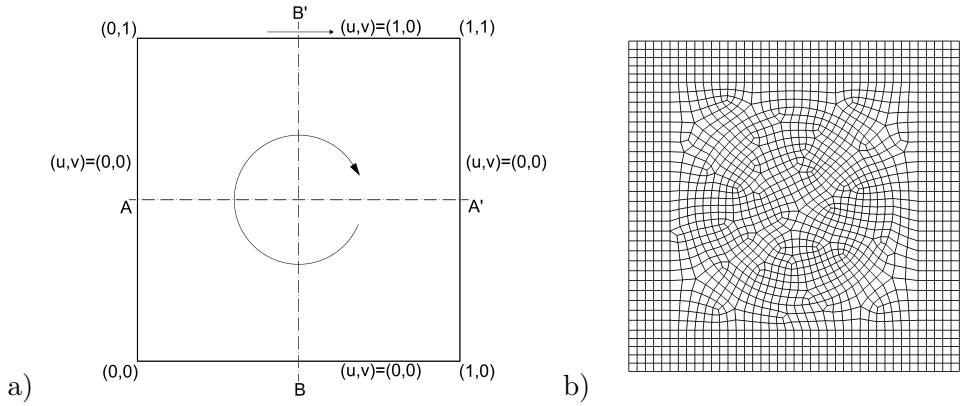


Figure 4.10: a) Lid-driven cavity flow configuration and boundary conditions. b) Unstructured mesh of 1635 quadrilateral cells used for the driven cavity flow problem.

In this work, the Reynolds number ranges from 100 to 1000. Figure 4.11 shows the solution obtained for the $Re = 1000$ case with the fourth-order FV-MLS method. The streamlines plotted in Figure 4.11 d) have a good agreement with the streamline pattern shown in [64].

Figures 4.12 and 4.13 present the velocity profiles obtained along the lines $x = 0.5$ and $y = 0.5$. The u velocity profiles are obtained over the line B-B' and the v velocity profile along the line A-A', as depicted in Figure 4.10.

For comparison, the problem was solved using the commercial software ANSYS-FLUENT 14.0. The results obtained with the commercial software are compared in Figure 4.12 with the numerical solutions obtained with the FV-MLS and the numerical solution from Ghia [64] computed on a 128×128 mesh, which is widely used by many authors [54, 96, 135] as a reference solution for this problem. The velocity profiles for $Re = 1000$ on the 1635 control volumes unstructured mesh are shown in Figure 4.12 for different orders of accuracy.

As shown in Figure 4.12, high-order discretizations offer the potential to increase the accuracy for a given mesh. When the third-order method is employed, the solution is very similar to the solution obtained by Ghia [64] on a structured mesh. The numerical solutions obtained with the FV-MLS are more accurate than the solutions obtained with

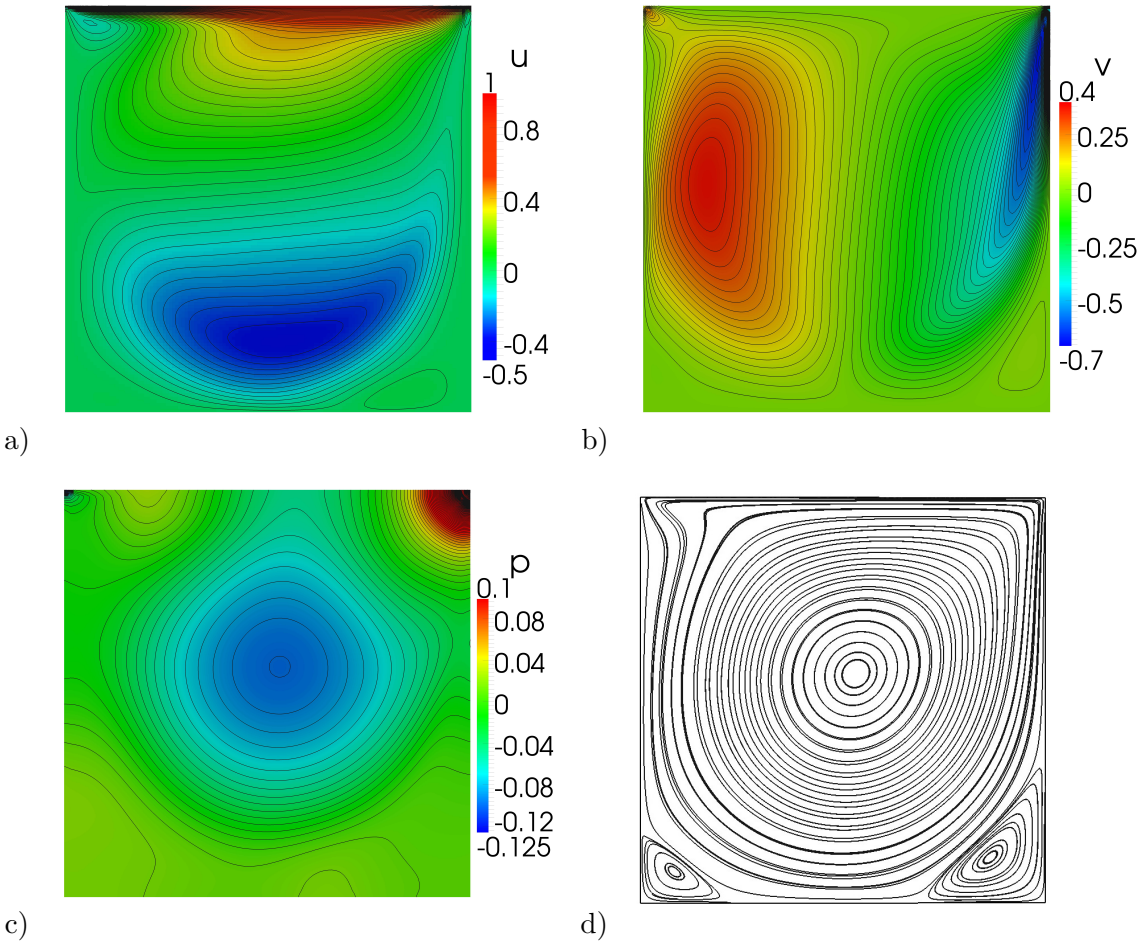


Figure 4.11: Isocontours of the u velocity field a), the v velocity field, b), the pressure field c) and streamlines d). Numerical solution of the Cavity flow at $Re = 1000$ and computed using the fourth-order FV-MLS scheme.

the commercial software ANSYS/FLUENT 14.0.

Next, the obtained results using the fourth-order FV-MLS method at $Re = 100$, 400 and 1000 are compared with the reference solution of Ghia [64] computed on a 128×128 mesh which is widely used by many authors [54, 96, 135] as a reference solution for this problem. The comparison presented in Figure 4.13 shows an excellent agreement with the reference solution for different Reynolds numbers. We only notice minor discrepancies on the v-velocity component at $Re = 1000$.

Note that fourth-order FV-MLS solution is obtained in an approximately 10 times coarser grid than the reference solution.

As shown in Figure 4.13, the results obtained for different Reynolds numbers on

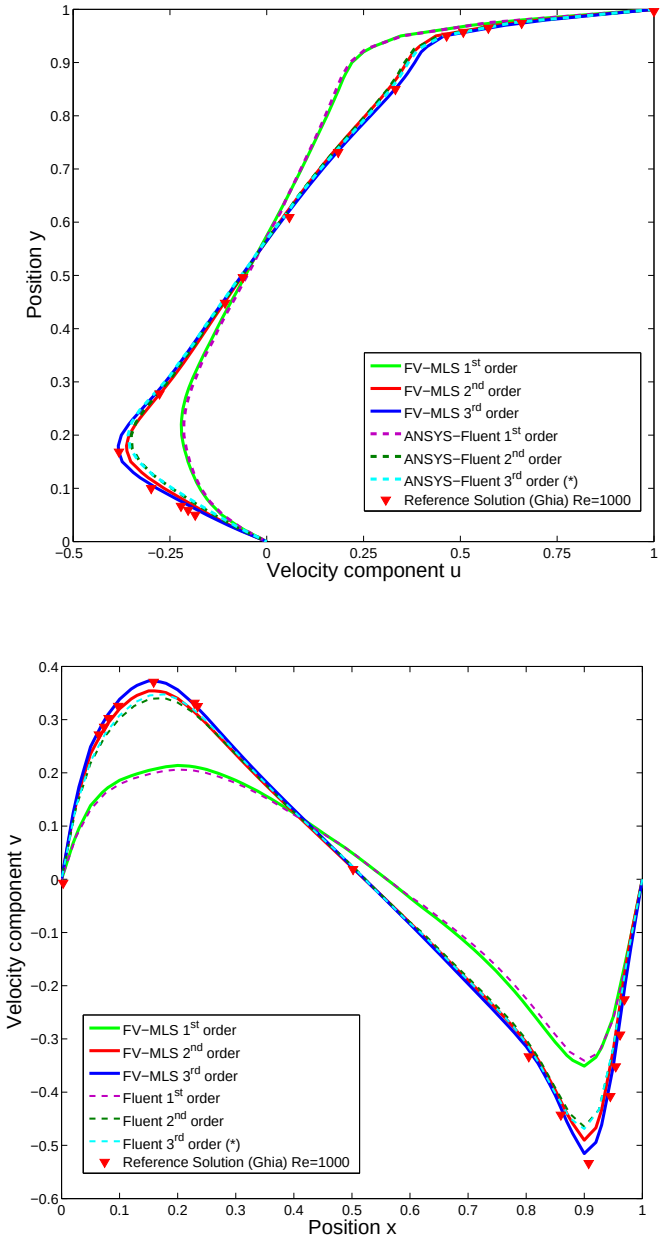


Figure 4.12: Comparison of the velocity profiles for different numerical orders of accuracy obtained on a 1635 control volumes unstructured mesh at $Re = 1000$. The numerical solutions were obtained with the FV-MLS method and the commercial software ANSYS/FLUENT 14.0.(*) The solution is obtained with third-order in the convective velocity and second-order in pressure.

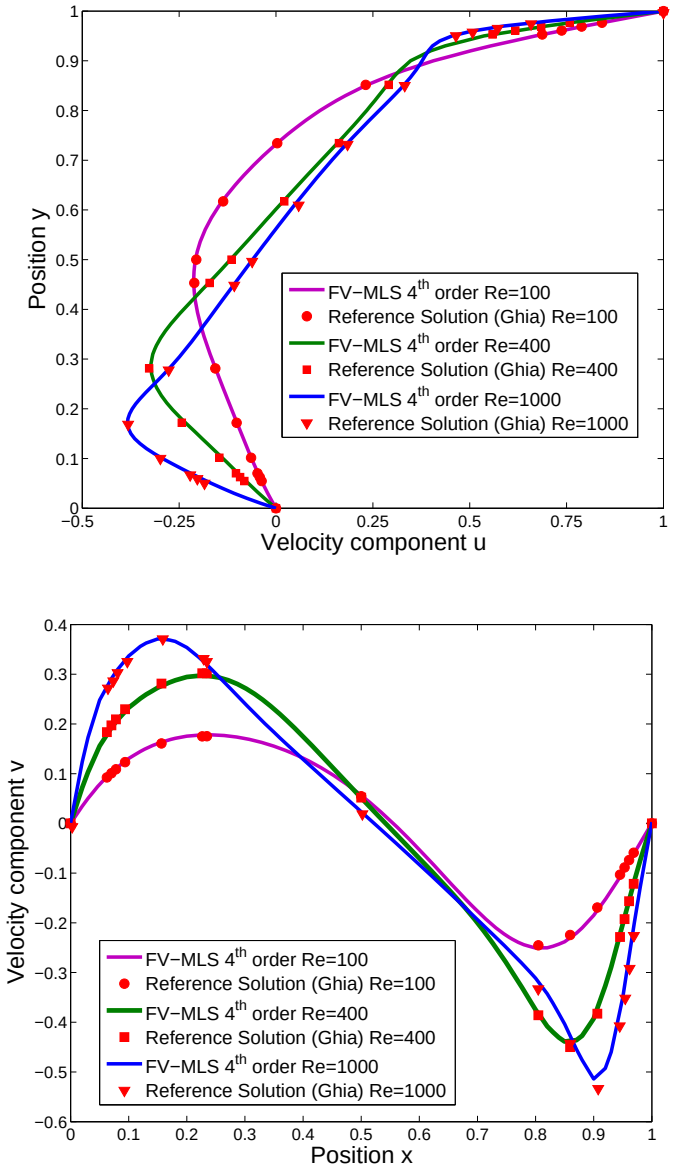


Figure 4.13: Velocity profiles at different Reynolds number obtained on a 1635 control volumes unstructured mesh using the fourth-order FV-MLS scheme.

unstructured meshes demonstrate the high-accuracy of the described new formulation.

4.4.4 Laminar Flow around a Cylinder

In this section, the laminar flow around a circular cylinder inside a channel is studied at $Re = 20$ and $Re = 100$. This flow is widely used as a benchmark to examine

the accuracy and behavior of numerical methods for incompressible flows [184, 221]. The problem setup was proposed by Schäfer and Turkel in [184]. The computational domain is presented in Figure 4.14. Note that the center of the cylinder is not on the horizontal symmetry plane of the channel. The diameter of the cylinder is $D = 0.1$ and the kinematic viscosity is set as $\nu = 10^{-3}$. The inflow velocity profile is given by

$$u(0, y) = \frac{4U_m y(H - y)}{H^2}, v(0, y) = 0 \tag{4.48}$$

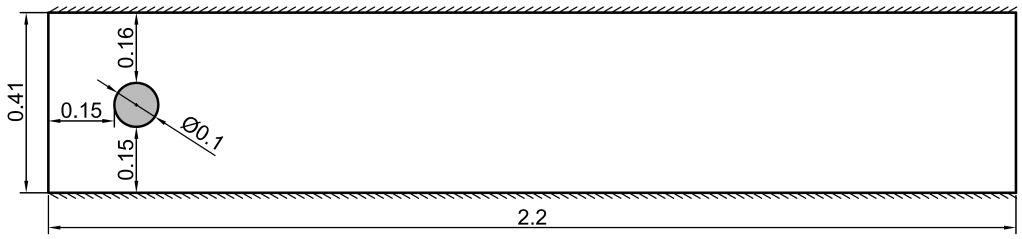


Figure 4.14: Geometry description for the laminar flow around a cylinder

where H is the height of the domain and U_m is the maximum velocity of the parabolic profile. Pressure outflow is prescribed at the outlet and no-slip boundary conditions are set on all remaining boundaries. The Reynolds number is defined by $Re = \overline{u}D/\nu$, where the value of the mean velocity is defined as $\overline{u} = \frac{2U_m}{3}$.

Two cases are investigated:

The first case is the steady computation of the laminar flow at $Re = 20$ which corresponds to test case 2D-1 defined by Schäfer [184]. For this case, the maximum velocity is set at $U_m = 0.3$. The benchmark parameters for the steady case are the drag (C_D) and lift (C_L) coefficients on the cylinder, the pressure difference (Δp) between the front and the back of the cylinder and the length of recirculation bubble (L_a). Table 4.7 presents the results obtained for different orders of accuracy and two different unstructured meshes. The coarsest grid is denoted by mesh A and has 4968 quadrilateral elements and mesh B has 19079 quadrilateral elements. The two different meshes used for this example are presented in Figure 4.15. An excellent agreement can be observed between the values given in [184] and the FV-MLS computations. In particular, the obtained results on the coarser grid ($N_{cell} = 4968$) are dramatically improved by using a higher-order discretization scheme. The use of a third-order accurate scheme on a finer grid is required to give a close agreement of Δp with the reference results. In Figure 4.16, the streamline pattern around the cylinder is obtained with the third-order scheme on the finer grid. Note that the streamline pattern

in Figure 4.16 is not symmetric due the asymmetry in the geometric configuration shown in Figure 4.14.

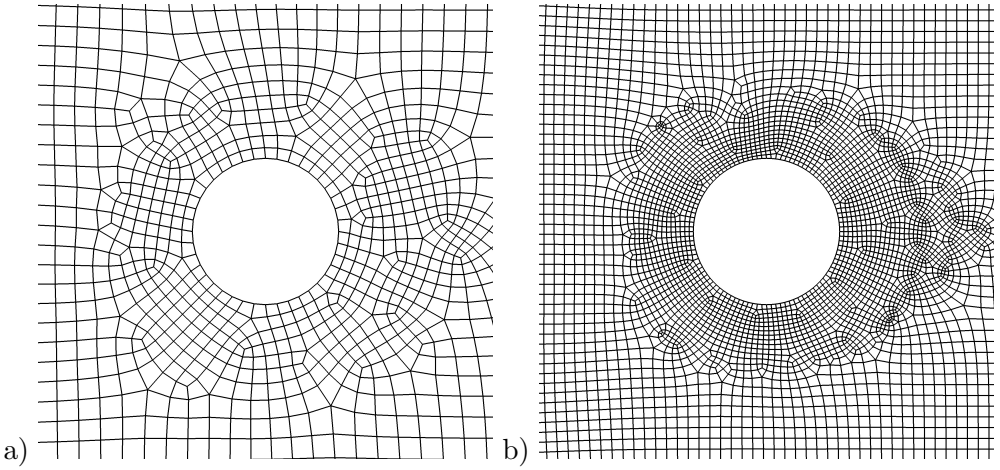


Figure 4.15: Unstructured meshes used for the FV-MLS computations of the Laminar Flow around a Cylinder. The coarsest mesh (a) is denoted by mesh A and has 4968 quadrilateral elements and the finest mesh (b) is denoted by mesh B and has 19079 quadrilateral elements.

Mesh	Order	C_D	C_L	L_a	Δp
Mesh A	2	5.5869	0.0087	0.0881	0.1149
(4968 cells)	3	5.5919	0.0108	0.0851	0.1161
Mesh B	2	5.5817	0.0113	0.0851	0.1168
(19079 cells)	3	5.5859	0.0107	0.0845	0.1174
Upper bound [184]	–	5.5900	0.0110	0.0852	0.1176
Lower bound [184]	–	5.5700	0.0104	0.0842	0.1172

Table 4.7: Comparison of the drag and lift coefficients, C_D and C_L, the increment of pressure Δp and the length of recirculation, L_a for different orders of accuracy and meshes.

The second case is the unsteady computation of the flow around a cylinder at $Re = 100$, namely case 2D-2 in [184]. The maximum velocity is defined as $U_m = 1.5$. The following quantities are computed: maximum drag coefficient (C_{Dmax}), maximum lift coefficient (C_{Lmax}), pressure difference (Δp) and the Strouhal number defined as $St =$

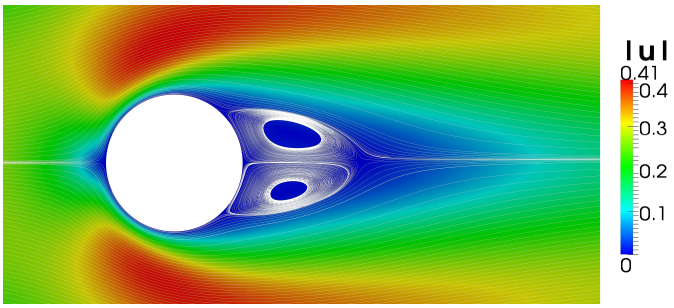


Figure 4.16: Velocity field and streamline pattern of the steady closed wake around a cylinder at $Re = 20$ obtained using third order FV-MLS scheme on the finer mesh.

$Df/\overline{u}$, where f is the vortex shedding frequency.

Grid	Order	C_{Dmax}	C_{Lmax}	St	Δp
Mesh A (4968 cells)	2	3.2741	1.2246	0.2825	2.3548
	3	3.2986	1.0451	0.2924	2.3962
Mesh B (19079 cells)	2	3.2702	1.0662	0.2952	2.4731
	3	3.2380	0.9985	0.3008	2.4858
Upper bound [184]	–	3.2400	1.0100	0.3050	2.5000
Lower bound [184]	–	3.2200	0.9900	0.2950	2.4600

Table 4.8: Unsteady flow around a cylinder at $Re = 100$. Comparison of the maximum drag and lift coefficients, C_{Dmax} and C_{Lmax}, the Strouhal number and the increment of pressure Δp for different orders of accuracy of the FV-MLS method.

On Table 4.8, the FV-MLS prediction of the global flow quantities are systematically improved by the use of third-order discretizations compared to second-order schemes. Figure 4.17 represents the pressure field and contours and Figure 4.18 shows the vorticity obtained on Mesh B with the third-order FV-MLS scheme.

4.5 Conclusions

On a finite volume framework, there are two kinds of grid arrangements used to solve the incompressible Navier-Stokes equations: staggered and collocated grids arrange-

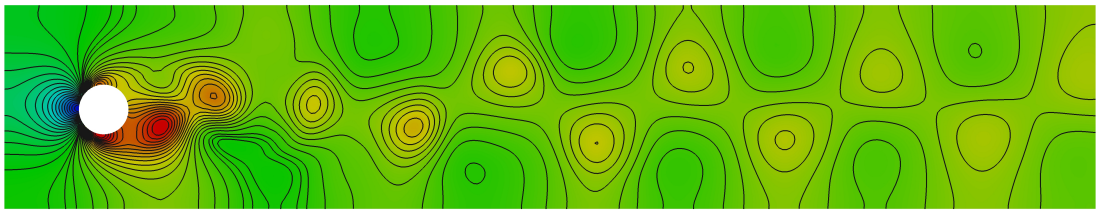


Figure 4.17: Pressure field and contours around a cylinder at $Re = 100$ obtained using third-order FV-MLS scheme

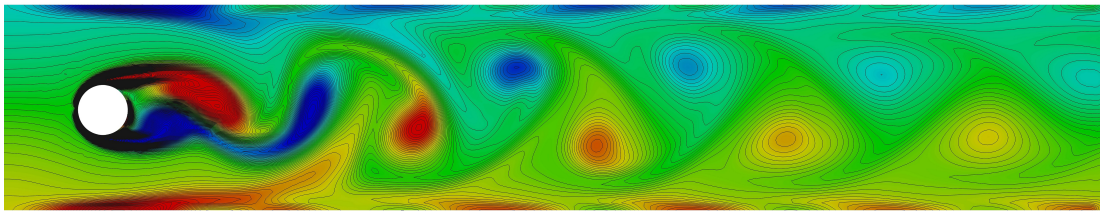


Figure 4.18: Vorticity field around the cylinder at $Re = 100$ obtained using third-order FV-MLS scheme

ment. The staggered grid arrangement presents some advantages, such as no interpolation is required due the variable arrangement, but its application to unstructured and curvilinear grids is not straightforward. On the other hand, the collocated grid arrangement is suitable for the resolution on unstructured grids. However, the collocated grid arrangement needs special treatment for the coupling of velocity and pressure and presents the possibility of checkerboard oscillations due to the central-difference discretization of the pressure.

The great geometrical flexibility offered by unstructured grids makes them highly effective for dealing with complex geometries, that are more typical of practical engineering applications. In this context, the development of very accurate numerical methods to work on unstructured grids is very desirable.

In this Chapter, a new higher-order accurate finite volume formulation for the numerical resolution of the incompressible Navier-Stokes equations. The novel formulation is based on a collocated grid arrangement, which is suitable for the numerical solution of fluid flows on unstructured meshes. In order to avoid checkerboard oscillations, a high-order extension of the Momentum Interpolation Method, presented by Rhie and Chow [174], is developed based on the Moving Least Squares approximants.

Incompressibility condition is imposed iteratively each time step using the SIMPLE algorithm. The Moving Least Squares approach is employed in order to obtain higher-order approximations and derivatives that allows us to obtain a higher-order numerical method

on unstructured collocated grids.

The proposed method is validated in several steady and unsteady test cases, where the accuracy and efficiency of the proposed numerical method is confirmed.

Numerical results on the Kovasznay and the 2D Taylor-Green Flow show that the new method gives the formal order of accuracy when linear, quadratic and cubic basis functions are used in the Moving Least Squares approximants. The laminar flow around a cylinder confined in a channel and the lid-driven cavity flow are studied and compared with the results obtained with other high-order schemes. The novel FV-MLS method for incompressible flows obtains more accurate results at lower mesh resolutions than standard second order methods.

In Chapters 3 and the present Chapter new high-order numerical methods for the resolution of the Navier-Stokes for all-speed and incompressible flows have been developed, analyzed and validated with several steady and unsteady test cases. In order to numerically simulate the flow on a turbomachine, the relative motion of the blades needs to be taken into account. In order to preserve the accuracy of the numerical scheme, the simulation of rotating geometries needs to be of, at least, the same order than the numerical scheme. This is a crucial point in the development of high-order methods for the simulations of turbomachines, and it will be addressed in Chapter 5, where new high-order Sliding mesh methods are developed.

Chapter 5

Moving grids. Development of high-order sliding-mesh techniques.

"Eppur si muove."
- Galileo Galilei.

5.1 Introduction

Flow unsteadiness and noise generation are currently among the most important limitations for the design of turbomachinery and renewable energy devices. These configurations involve complex unsteady flow patterns like vortex shedding, stalled flows, blade wake interactions which are, in general, due to the presence of moving or oscillating bodies. On one hand, one must employ high-order numerical methods to accurately compute both the unsteady flow field, as the formulations developed in Chapters 2 and 3. On the other hand, dedicated techniques must be employed to carefully deal with the interface between static and moving grids in an unsteady flow framework. Such issue can be addressed using several numerical approaches, among others, the phase-lagged periodic boundary conditions for rotor-stator interaction in axial compressor [51, 74, 75], the body-fitted approach in an Arbitrary Lagrangian Eulerian (ALE) setting, Cartesian grid methods based on the immersed-boundary[163] or on the cut cell methods [13, 228] and the non-boundary conforming sliding mesh approach. The later is attractive due to its ability to capture flow unsteadiness without requiring the use of a filtering procedure nor computationally expensive re-meshing strategies. The sliding mesh method was successfully employed by Rai et al. [167, 168] for the computation of rotor-stator interactions in a supersonic flow. This patched-grid technique allows relative sliding of one mesh adjacent to another static or

moving mesh. A three steps explicit zonal scheme, which preserve flux conservation at the interface, is proposed in [166]. More recently, Gourdain et al. [68] employed the sliding mesh approach for the simulation of large-scale industrial flows in multistage compressors. In they comparative study between Chimera and sliding mesh techniques for unsteady simulations of counter rotating open-rotors, Francois et al. [60] shown that these methods give similar accuracy but the later requires much less memory than the Chimera approach. Note also that the sliding mesh algorithm was used by Steijl and Barakos [194] for the computational fluid dynamic analysis of helicopter rotor-fuselage aerodynamics.

Nowadays, sliding mesh techniques are commonly used to compute non-axisymmetrical unsteady flow fields and corresponding aerodynamic performances of cross-flow fans [138, 204] and wind turbines [10, 89, 91, 92, 95, 99, 100, 113]. In particular, McNaughton et al. [131] obtained a good agreement between coupled LES-sliding interfaces for thrust and power predictions of a tidal-stream turbine. As far as aeroacoustic computations are concerned, Moon et al. [138] developed a time-accurate viscous flow solver for the prediction of unsteady flow characteristics and the associated aeroacoustic blade tonal noise of a cross flow impeller. The sliding mesh approach, which was implemented in an unstructured finite-volume solver on triangular meshes, was able to correctly predict the oscillations of the eccentric vortex due to the mismatch of blade incidence angles and the recirculation bubbles around the blades.

However, most of the sliding mesh methods proposed in the literature belong to the family of at most second-order interpolation schemes. Therefore they can not be used in conjunction with higher-order numerical schemes without depreciating the overall accuracy of the numerical methods.

To the author knowledge, few studies addressed such problem. A high order ($order \geq 3$) h/p Discontinuous Galerkin method with sliding mesh capabilities was recently proposed by Ferrer and Willden [58] for the computation of the unsteady incompressible flow field of a three bladed cross-flow turbine. They successfully obtain spectral convergence rate when solving the incompressible Navier-Stokes equations on non-conformal grids. In [11] a mesh moving technique for sliding interfaces is presented for the numerical simulation of a wind turbine with a FEM-based ALE-VMS (variational multiscale formulation written in the arbitrary Lagrangian-Eulerian frame) formulation.

In this Chapter the relative motion of the blades is taken into account through the development new high-order sliding mesh methods based on Moving Least Square method explained in Chapter 2. The novel methods fits in the framework of high-order FV-MLS schemes developed in the previous Chapters, and they will be analyzed and validated in several compressible and incompressible test cases.

The Chapter is organized as follows. In section 2, the governing equations are written. The new MLS-based sliding-mesh technique is presented in section 4. Then, section 5 is devoted to numerical simulations. Finally, the conclusions are drawn in section 6.

5.2 Governing equations and numerical methods

In order to account the relative mesh motion of one mesh with respect to other, it is advantageous to write the two-dimensional compressible Navier-Stokes equations, already defined in equation (1.1) in the Arbitrary Lagrangian-Eulerian (ALE) form,

$$\frac{\partial \boldsymbol{U}}{\partial t}+\frac{\partial\left(\boldsymbol{\mathcal{F}}_{\boldsymbol{x}}^{\boldsymbol{\mathcal{H}}}-\boldsymbol{\mathcal{F}}_{\boldsymbol{x}}^{\boldsymbol{\mathcal{V}}}\right)}{\partial x}+\frac{\partial\left(\boldsymbol{\mathcal{F}}_{\boldsymbol{y}}^{\boldsymbol{\mathcal{H}}}-\boldsymbol{\mathcal{F}}_{\boldsymbol{y}}^{\boldsymbol{\mathcal{V}}}\right)}{\partial y}=\mathbf{0} \tag{5.1}$$

where $\boldsymbol{U}$ is the vector of variables $\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}}=\left(\boldsymbol{\mathcal{F}}_{\boldsymbol{x}}^{\boldsymbol{\mathcal{H}}}, \boldsymbol{\mathcal{F}}_{\boldsymbol{y}}^{\boldsymbol{\mathcal{H}}}\right)$ is the inviscid flux vector and $\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{V}}}=\left(\boldsymbol{\mathcal{F}}_{\boldsymbol{x}}^{\boldsymbol{\mathcal{V}}}, \boldsymbol{\mathcal{F}}_{\boldsymbol{y}}^{\boldsymbol{\mathcal{V}}}\right)$.

For compressible flows the conservatives variables are defined as

$$\boldsymbol{U}(\boldsymbol{x}, t)=\left\{\begin{array}{c}\rho \\ \rho v_x \\ \rho v_y \\ \rho E\end{array}\right\}. \tag{5.2}$$

and the inviscid fluxes are given by

$$\boldsymbol{\mathcal{F}}_{\boldsymbol{x}}^{\boldsymbol{\mathcal{H}}}=\begin{pmatrix}\rho v_x-\rho v_x^{mesh} \\ \rho v_x^2+p-\rho v_x v_x^{mesh} \\ \rho v_x v_y-\rho v_y v_x^{mesh} \\ \rho v_x H-\rho E v_x^{mesh}\end{pmatrix} \qquad \boldsymbol{\mathcal{F}}_{\boldsymbol{y}}^{\boldsymbol{\mathcal{H}}}=\begin{pmatrix}\rho v_y-\rho v_y^{mesh} \\ \rho v_x v_y-\rho v_x v_y^{mesh} \\ \rho v_y^2+p-\rho v_y v_y^{mesh} \\ \rho v_y H-\rho E v_y^{mesh}\end{pmatrix} \tag{5.3}$$

where the (u_{mesh}, v_{mesh}) is the mesh velocity.

The viscous fluxes $\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{V}}}$ are given by the following expression,

$$\boldsymbol{\mathcal{F}}_{\boldsymbol{x}}^{\boldsymbol{\mathcal{V}}}=\begin{pmatrix}0 \\ \tau_{xx} \\ \tau_{xy} \\ v_x \tau_{xx}+v_y \tau_{xy}-q_x\end{pmatrix} \quad \boldsymbol{\mathcal{F}}_{\boldsymbol{y}}^{\boldsymbol{\mathcal{V}}}=\begin{pmatrix}0 \\ \tau_{xy} \\ \tau_{yy} \\ v_x \tau_{xy}+v_y \tau_{yy}-q_y\end{pmatrix} \tag{5.4}$$

The viscous stresses are modeled as

$$\begin{aligned}
\tau_{xx} &= 2\mu\frac{\partial v_x}{\partial x} - \frac{2}{3}\mu\left(\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y}\right) \\
\tau_{yy} &= 2\mu\frac{\partial v_y}{\partial y} - \frac{2}{3}\mu\left(\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y}\right) \\
\tau_{xy} &= \mu\left(\frac{\partial v_x}{\partial y} + \frac{\partial v_y}{\partial x}\right)
\end{aligned} \tag{5.5}$$

For incompressible flows, the assumption of incompressibility lead us to a system of equations with the following variables

$$\boldsymbol{U}(\boldsymbol{x},t) = \left\{ \begin{array}{c} 0 \\ \rho v_x \\ \rho v_y \end{array} \right\}. \tag{5.6}$$

The inviscid fluxes are

$$\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}}_{\boldsymbol{x}} = \begin{pmatrix} \rho v_x - \rho v_x^{mesh} \\ \rho v_x^2 + p - \rho v_x v_x^{mesh} \\ \rho v_x v_y - \rho v_y v_x^{mesh} \end{pmatrix} \qquad \boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}}_{\boldsymbol{y}} = \begin{pmatrix} \rho v_y - \rho v_y^{mesh} \\ \rho v_x v_y - \rho v_x v_y^{mesh} \\ \rho v_y^2 + p - \rho v_y v_y^{mesh} \end{pmatrix} \tag{5.7}$$

The viscous fluxes are given by

$$\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{V}}}_{\boldsymbol{x}} = \begin{pmatrix} 0 \\ \mu\dfrac{\partial v_x}{\partial x} \\ \mu\dfrac{\partial v_y}{\partial x} \end{pmatrix} \quad \boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{V}}}_{\boldsymbol{y}} = \begin{pmatrix} 0 \\ \mu\dfrac{\partial v_x}{\partial y} \\ \mu\dfrac{\partial v_y}{\partial y} \end{pmatrix} \tag{5.8}$$

5.3 General formulation

Following equation (2.8), the Navier Stokes equations are written in the discrete form

$$\Omega_I \frac{\partial u_I}{\partial t} = \sum_{j=1}^{N_f}\sum_{ig=1}^{N_G}\left[\left(\boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{V}}} - \boldsymbol{\mathcal{F}}^{\boldsymbol{\mathcal{H}}}\right)_j \cdot \hat{\boldsymbol{n}}_j\right]_{ig} \mathcal{W}_{ig} \tag{5.9}$$

The numerical discretization of equation (5.9) was explained in Chapter 3 for compressible flows and Chapter 4 for incompressible.

It is important to remark that for a stationary mesh the construction of the discretization stencil and the computation of the MLS shape function derivatives are done as

a preprocessing step prior to the iterative procedure. As a consequence, the extra computational time relative to the high-order FV-MLS scheme compared to second order scheme is mainly due to the loop over the quadrature points required for the high-order flux integration. If the mesh is not stationary, for example when using the sliding mesh approach, the MLS shape functions need to be recomputed but only in those elements where the stencil has been modified due to mesh displacement.

5.4 MLS-based sliding mesh

The sliding mesh technique requires two meshed zones in relative motion between them. Even though the methodology presented here is valid for any pair of grids in relative motion, for clarity this section will focus on the case of one moving grid *sliding* over one fixed grid. This is, for example, the case of turbomachinery, where one of the grids is related to the stator (fixed) and the other is related to the rotor (moving). The sliding takes place on a plane that is called *interface*. Figure 5.1 schematically shows this idea. Note that in practice, the interface is composed by two coincident edges. One of them belongs to the fixed grid and the other to the moving mesh.

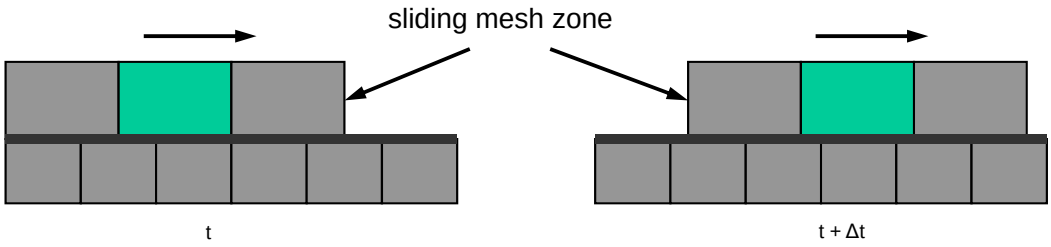


Figure 5.1: Schematic illustration of the sliding mesh concept. The top zone slides over the fixed grid. Note the non-conformal grid.

As the moving grid slides over the fixed grid, the mesh is no longer conformal.

Hereafter, the nodes/cells places at the interface will be denoted as *intersection nodes/cells*, as shown in Figure 5.1. First, the identification of the set of neighbors for each of the intersection cells, namely the computational stencil, must be performed in order to evaluate the MLS-shape functions. To this end, both full stencil and half stencil are considered. The former takes neighboring cells from both grids as shown by shaded triangles in Figure 5.2. The latter is simply made with cells of the same grid.

Second, we must develop efficient and robust MLS-based sliding mesh interfaces for the transmission of information for one grid to another. The proposed different techniques

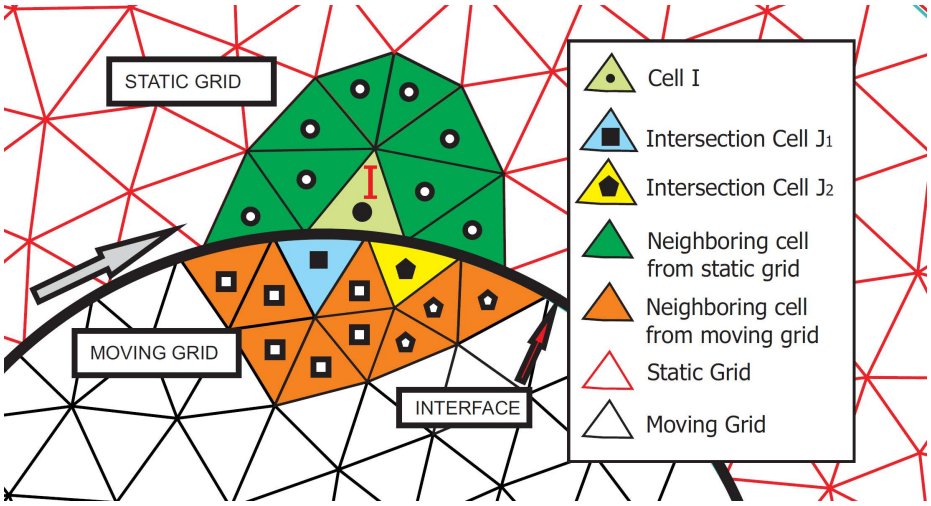


Figure 5.2: Schematic illustration of stencil for MLS interpolation at interface.

belong to the two following sliding mesh families, namely the *intersection-based* and the *interface-halo-cell* approaches.

Therefore, the three high-order approaches, which are presented in sections 5.4.1 and 5.4.2, will be referred hereafter as

1. Full Stencil with Intersections (FS Intersections)
2. Half Stencil with Intersections (HS Intersections)
3. Full Stencil with Halo cell (FS Halo)

5.4.1 MLS-based sliding mesh with intersections

This approach requires the computation of intersections at the interface. The novelty of this approach is the use of MLS to obtain a high-order accurate reconstruction. Let *main interface* be the interface edge that is part of the moving mesh, and *secondary interface* the interface edge that belongs to the fixed mesh (see figure 5.3).

First, the main interface edges are identified by performing a loop over the edges of the moving mesh. If one edge is labeled as interface edge, the secondary interface is straightforwardly determined. Once the main and secondary interfaces are determined, intersection nodes can be found.

The exchange of information between the moving domain and the fixed one is performed at both main and secondary interfaces. Since the grid is not conformal at the interface between moving and fixed domains, the corresponding numerical fluxes must be computed for each cells sharing the face of the neighboring interface cell. This procedure is

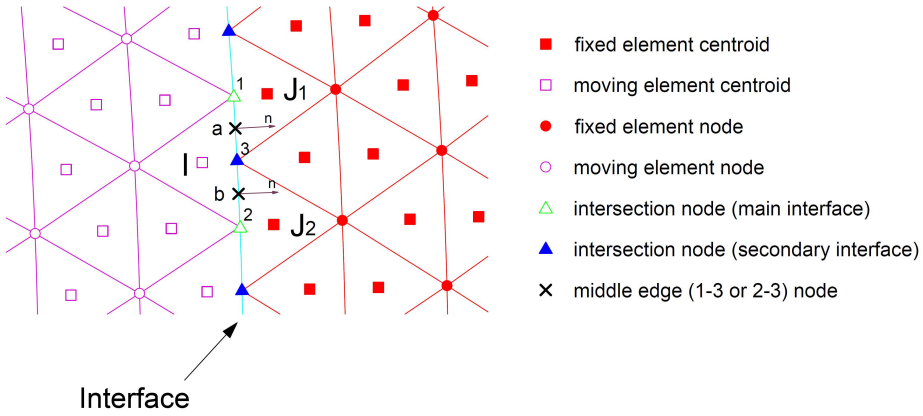


Figure 5.3: Schematic representation of intersection nodes.

illustrated in Figure 5.4 where the flux exchange between cells I/J_1 and cells I/J_2 must be computed.

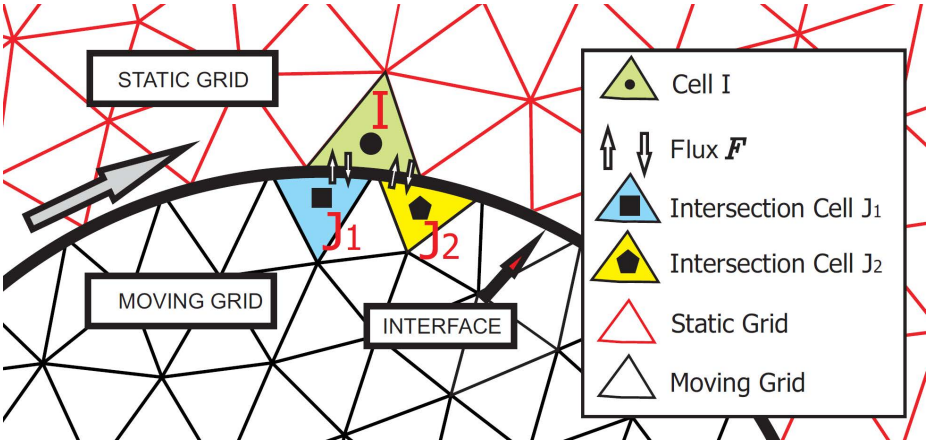


Figure 5.4: Schematic representation of the flux exchange for the MLS-based sliding mesh with intersections approach.

Full stencil with intersections (FS Intersections)

In the full stencil approach, the stencil of cell I is built using the union of the stencils of the cell I and all the fixed elements of the secondary interface. Therefore, the total stencil of cell I is computed by merging stencils of cell I and those cells having an interface edge coincident with cell I (shaded in orange in Figure 5.2). The final stencil of cell I is shown in Figure 5.5.

Numerical fluxes at interface are computed, as usual, at integration points of

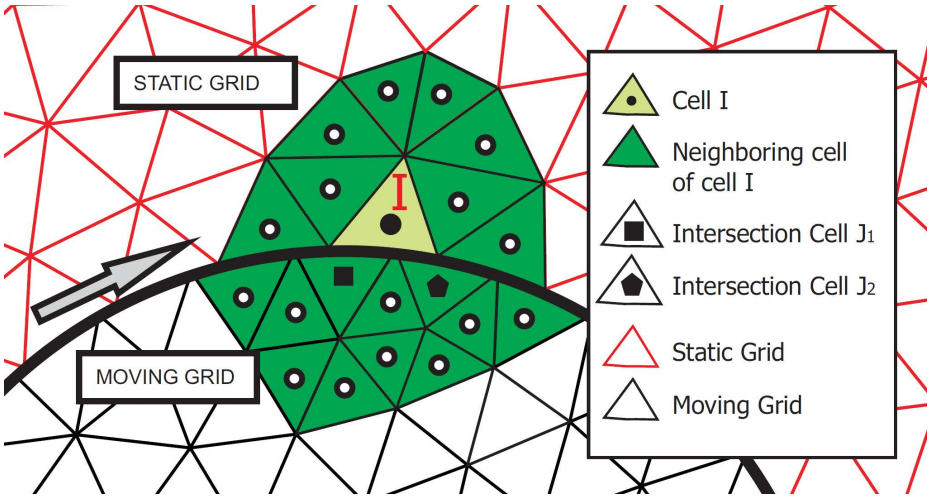


Figure 5.5: Schematic representation of the Full Stencil approach.

each edge. This procedure must be performed each time step. In problems related with rotor/stator configurations, the location of interface and the numeration of the cells next to it in both domains, fixed and moving is known a priori. The computational cost associated to the evaluation of MLS shape functions at each time-steps is limited since the cells requiring re-computation of the MLS shape functions are those near the interface. This additional cost greatly depends on the grid topology and the level of complexity of the configuration.

Half stencil with intersections (HS Intersections)

In order to avoid both the search of neighbors and the computation of the MLS shape functions at each time step, the use of a biased stencil is proposed, taking only in account cells from the grid in which the cell is placed. This procedure is depicted in Figure 5.6. Obviously, this half-stencil approach is expected to be less accurate than its full stencil counterpart since the computational stencil of cell I is not centered.

5.4.2 Interface-halo-cell sliding mesh

The need for computation of face/edge intersections is one of the drawbacks of previous approach. It therefore introduces additional complexity in the coding and also in the computing time. Here, the Interface-halo-cell stencil is introduced as an alternative to the intersection-based stencils described previously. Consider a cell I, a *halo* cell is created in front of this cell, as it is shown in Figure 5.7. The centroid is denoted as P_H and Ω_{P_H} is referred to the area of the halo cell. Note that the halo cell is a specular image of cell I.

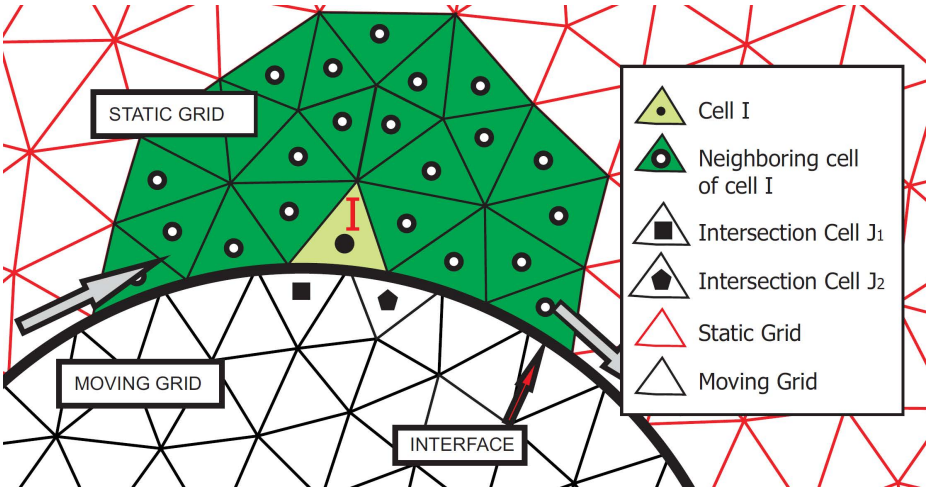


Figure 5.6: Schematic representation of the Half Stencil approach.

In case of taking in account the curvature of the interface, the areas of I and A_{P_H} will no longer be the same. The value of the variables at P_H is obtained as

$$\boldsymbol{U}_{P_H} = \frac{1}{\Omega_{P_H}} \int \boldsymbol{U} d\Omega = \frac{1}{\Omega_{P_H}} \int \sum_{k=1}^{n_x} N_k(\boldsymbol{x}_{P_H}) \boldsymbol{U}_k d\Omega \tag{5.10}$$

The value $\boldsymbol{U}^-$ at the integration point is then computed by a Taylor approximation of $\boldsymbol{U}$ from P_H and then we compute the flux at cell I as usual. This is schematically presented in figure 5.7.

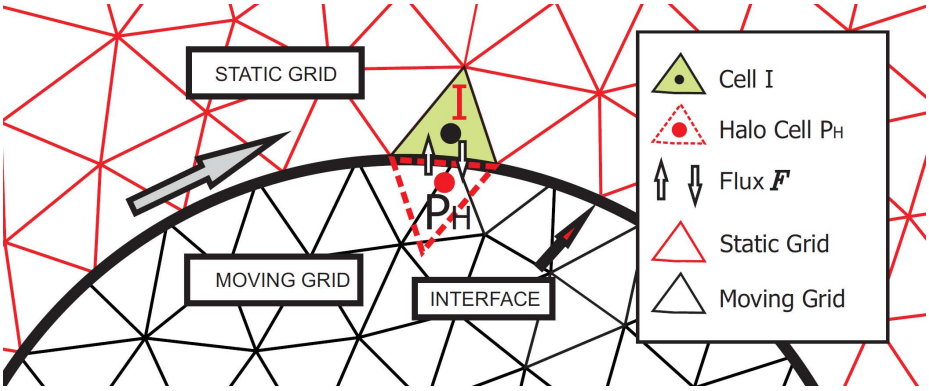


Figure 5.7: Schematic representation of Interface-halo-cell sliding mesh. Note that the flux can be computed without the definition of intersections

Halo cell computational stencil

In order to compute the value of the variables $\boldsymbol{U}$ at the centroid of the halo cell as in (5.10), there is a need to compute the MLS shape function $\boldsymbol{N}(\boldsymbol{x}_{\boldsymbol{P_H}})$ at this point. The high-order computation of the fluxes is achieved computing the derivatives of the shape function for the *halo* cell.

To this end, we need to obtain the stencils for cells I and P_H with cells from both regions (as shown in Figure 5.8). Once the stencils are defined, the computation of the MLS shape functions is performed.

Note that, similar to the intersection-based sliding mesh interfaces, the stencil and shape functions need to be updated each time that the grid moves, but no intersections have to be computed.

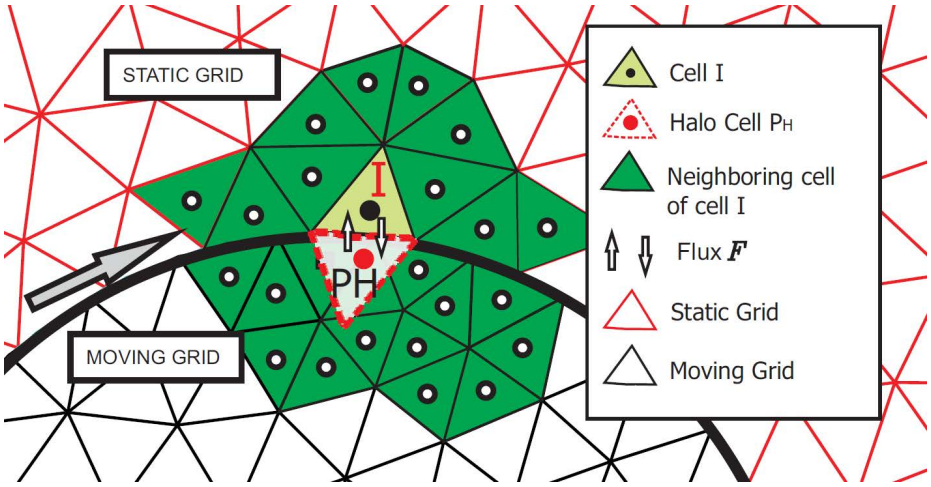


Figure 5.8: Schematic representation of Interface-halo-cell sliding mesh with full stencil

Conservation properties of Halo cell interfaces

Although the halo cell approach is not conservative by nature, the aim of this section is to demonstrate that the conservation error is within the same order of magnitude than the error in variables, and the convergence order of the conservation error is at least the same than that of the numerical scheme.

To this end, the behavior of the Interface-halo-cell sliding mesh formulation is studied when a steady shock is located at the interface. This test case is taken from [223]. The flow is assumed to travel from left to right. The initial conditions on the left and right side of the shock wave are

$$\begin{aligned} \rho_L &= 1, & \rho_R &= 1.8621 \\ u_L &= 1.5, & u_R &= 0.8055 \\ p_L &= 0.71429, & p_R &= 1.7559 \end{aligned} \tag{5.11}$$

The full computational domain is $0 \leq x \leq 10$ and it is discretized in two regions of 25 elements each. The interface between the two regions is located at $x = 5.0$. Figure 5.9 shows a schematic view of the computational regions, the interface and the corresponding halo cell for each domain.

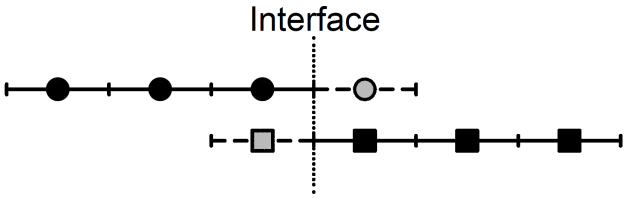
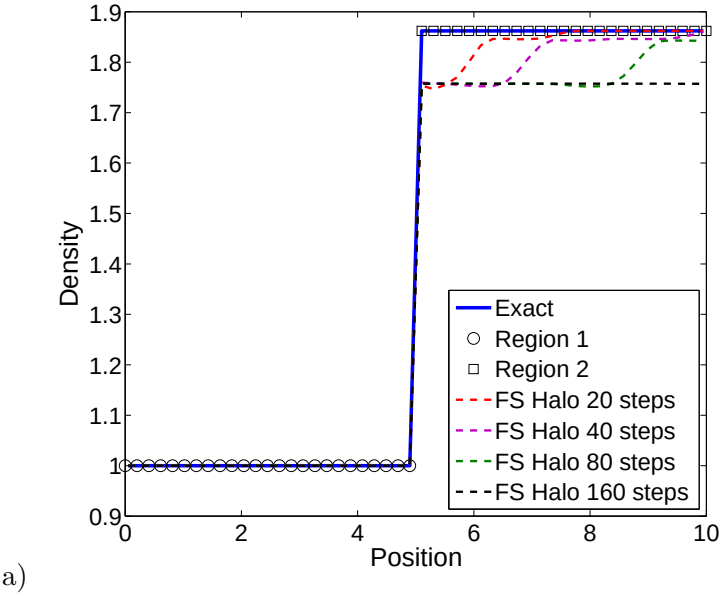


Figure 5.9: Schematic view of the computational domains at $x = 5.0$ with the corresponding halo cell (dashed line) for each domain at the interface.

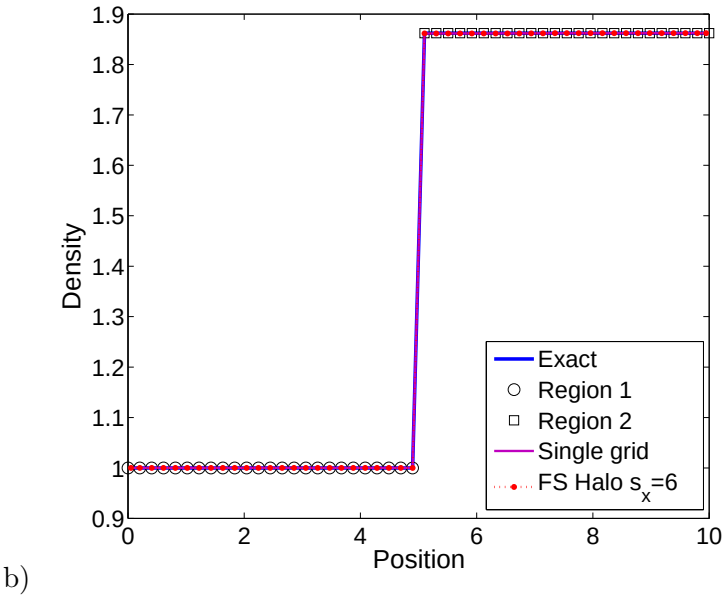
In Figure 5.10, the obtained results are shown with a single grid for the whole domain and with two grids using the halo cell formulation with full stencil. It is observed that when the parameter of the kernel is less than $s_x = 6$ a spurious solution is obtained. If the kernel parameter is set to $s_x = 6$ for the interface cells, the shock is preserved, as seen in Figure 5.10 b). The reason of this behavior lies on the nature of the interpolation. As seen in Figure 2.7, when $s_x < 6$ the shape of the kernel is not close enough to the delta function [149], and the value of the variables at the cells of the stencil far from the shock introduce an oscillation that propagates downstream.

In the second test we continue to assess the performance of the Interface-halo-cell sliding mesh formulation. We present an unsteady shock that moves through the interface. This test case corresponds to the Test 1 of [205]. The initial states are

$$\begin{aligned} \rho_L &= 1.0, & \rho_R &= 0.125 \\ u_L &= 0.75, & u_R &= 0.0 \\ p_L &= 1.0, & p_R &= 0.1 \end{aligned} \tag{5.12}$$



a)



b)

Figure 5.10: a) Time evolution of the numerical results of the 1D steady shock when the FS Halo cell approach is employed with $s_x = 4.5$. The interface is located at $x = 5$. A numerical oscillation propagation can be observed. After 160 steps the stationary state is reached but the shock is not preserved. b)Comparison of the numerical results of the 1D steady shock between the FS Halo cell approach with $s_x = 6.0$ and when a single mesh grid is employed to discretize the domain. No discrepancies are observed.

The domain $[0, 1]$ is discretized in two domains with 200 computing cells each and the interface is located at $x = 0.5$. Figure 5.11 presents a comparison of the density profile obtained with a single grid and with two different grids with the interface located at $x = 0.5$.

The comparison presented in Figure 5.11 shows an excellent agreement between the single grid solution and the halo-cell interface solution. Note that there is no any visible numerical artifacts at the interface ($x = 0.5$) when the halo cell formulation with full stencil and $s_x = 6$ is used.

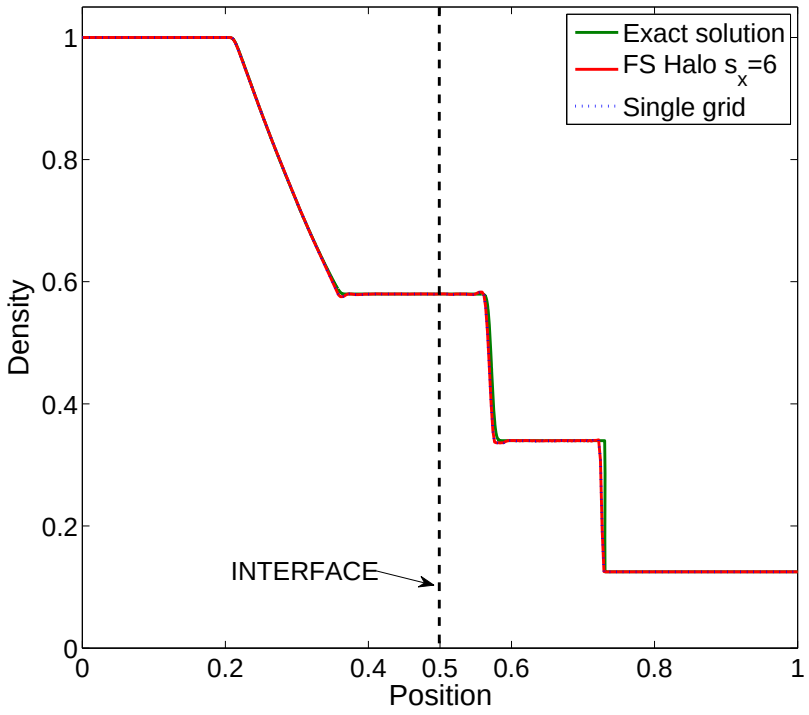


Figure 5.11: Numerical solutions at $t = 0.2$ for the 1D Unsteady Shock with 3^{rd} order FV-MLS method. The interface is located at $x = 0.5$.

5.5 Numerical Examples

This section presents numerical results for several test cases aimed at assessing the accuracy and efficiency of the proposed method for both steady and unsteady flow problems. The first three test cases study the accuracy and the conservation property of the method for non conformal meshes. The last example shows the application of the method to the solution of the incompressible Navier-Stokes equations.

5.5.1 Ringleb flow

In this 2D example the rate of convergence and the conservation error are studied for the different sliding mesh approaches derived in section 5.4. This test case, already explained in Chapter 1, is widely used as a benchmark for compressible codes [25, 148]. The flow is obtained as a solution of the hodograph equation. The transformation equations between the Cartesian variables (x, y) and the hodograph variables (V, θ) are described in [25].

The square domain $\Omega = [-1.15, -0.75] \times [0.15, 0.55]$ was discretized using four different unstructured mesh with non conformality at the interface of 580, 2270, 9044 and 35918 triangular elements.

Figure 5.12 shows the schematic description of the problem and the coarsest grid. The analytical solution is prescribed at the boundaries. In order to check the formal order of accuracy of our numerical solver, we compute the L_2 norm of the entropy error as

$$L_2^{ent} = \sqrt{\frac{1}{\Omega}\int_\Omega \left(\frac{p/\rho^\gamma - p_\infty/\rho_\infty^\gamma}{p_\infty/\rho_\infty^\gamma}\right)^2 d\Omega} \tag{5.13}$$

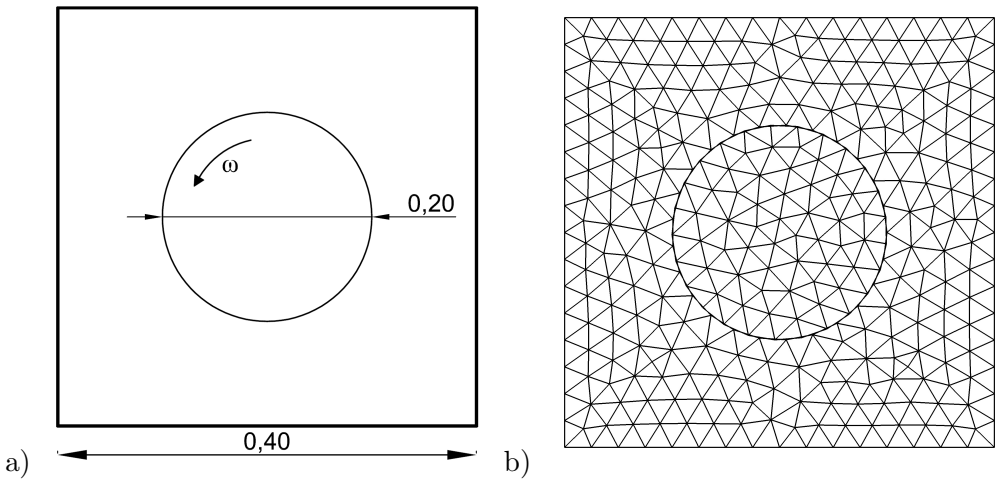


Figure 5.12: a) Geometry description of Ringleb flow problem. b) Unstructured mesh of 580 triangles. Note the non-conformal mesh at the interface

The conservation error is evaluated as the difference of sum of the fluxes from both sides of the interface

$$L_2^{cv} = \frac{1}{\Gamma_{Interface}}\int \left| \boldsymbol{\mathcal{F}^{\mathcal{H}}}_1 - \boldsymbol{\mathcal{F}^{\mathcal{H}}}_2 \right| d\Gamma_{Interface} \tag{5.14}$$

where $\Gamma_{Interface}$ denotes the perimeter of the interface, $\boldsymbol{\mathcal{F}^{\mathcal{H}}}_1$ the convective flux obtained at the interface of region 1 and $\boldsymbol{\mathcal{F}^{\mathcal{H}}}_2$ is obtained at the interface of region 2.

Tables 5.1 to 5.4 present the L_2 norm of the entropy error, the conservation error and corresponding orders of accuracy for 3rd and 4th order accurate FV-MLS schemes. For validation purpose, first, the investigation is carried out on non conformal static grids(Tables 5.1 and 5.2).

Third-order $\omega = 0$ rad/s					
Method	Mesh	L_2 Entropy	Order	Conservation Error	Order
FS Intersections	580	8.44×10^{-8}	-	-	-
	2270	9.05×10^{-9}	3.27	-	-
	9044	1.14×10^{-9}	3.00	-	-
HS Intersections	580	9.19×10^{-8}	-	-	-
	2270	9.83×10^{-9}	3.28	-	-
	9044	1.27×10^{-9}	2.97	-	-
FS Halo	580	9.19×10^{-8}	-	2.62×10^{-8}	-
	2270	9.29×10^{-9}	3.36	2.03×10^{-9}	3.75
	9044	1.19×10^{-9}	2.98	2.83×10^{-10}	2.85

Table 5.1: Accuracy orders, conservation error and L_2 norm of entropy error for the Ringleb flow test case for the third-order FV-MLS scheme employing the three different sliding mesh approaches on non-conformal meshes.

The three methods show the expected order of accuracy for the L_2 norm of the entropy error. For the third-order scheme all the three approaches give similar results in terms of accuracy, whereas for the fourth-order scheme the HS-Intersection approach is clearly less accurate. The conservation error of the halo-cell approach is below the entropy error, and it keeps the expected order of convergence.

Fourth-order $\omega = 0$ rad/s					
Method	Mesh	L_2 Entropy	Order	Conservation Error	Order
FS Intersections	580	3.76×10^{-9}	-	-	-
	2270	2.72×10^{-10}	3.85	-	-
	9044	1.80×10^{-11}	3.93	-	-
HS Intersections	580	1.21×10^{-8}	-	-	-
	2270	7.25×10^{-10}	4.12	-	-
	9044	4.00×10^{-11}	4.16	-	-
FS Halo	580	4.16×10^{-9}	-	3.48×10^{-9}	-
	2270	2.66×10^{-10}	4.03	2.97×10^{-11}	6.98
	9044	1.76×10^{-11}	3.93	2.60×10^{-12}	3.53

Table 5.2: Accuracy orders, conservation error and L_2 norm of entropy error for the Ringleb flow test case for the fourth-order FV-MLS scheme employing the three different sliding mesh approaches on non-conformal meshes.

These remarks are confirmed by the plot of the L_2 norm of entropy errors as a function of different mesh resolutions (Fig. 5.13). All sliding mesh approaches give very similar results for the 3rd order scheme and the lack of accuracy of HS-Intersection method is clearly visible for 4th order spatial discretizations.

Next, the same analysis is carried out with a rotational velocity of $\omega = 0.01$ rad/s. The results are plotted in Tables 5.3 and 5.4. Note that the accuracy of the sliding mesh methods is not depreciated compared to the static case and similar behavior is observed.

In Figure 5.14 the L_2 norm of the conservation error is plotted for $\omega = 0$ rad/s and $\omega = 0.01$ rad/s.

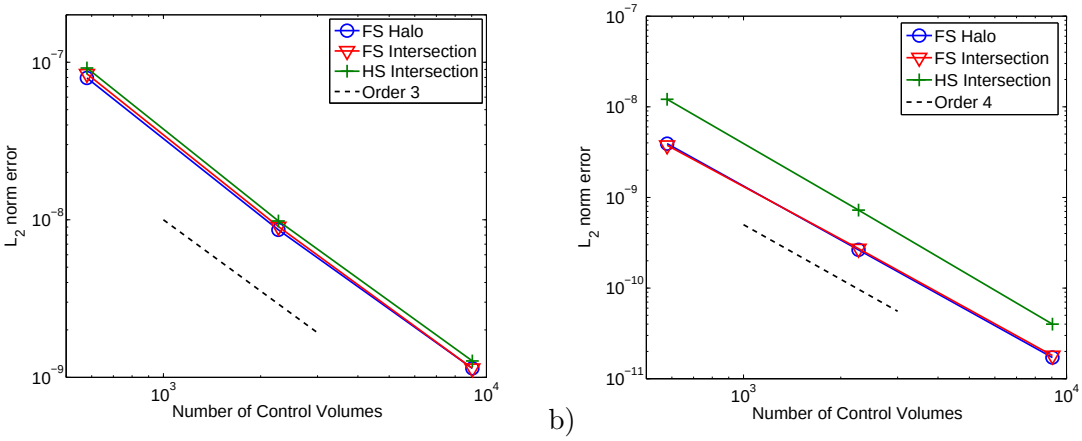


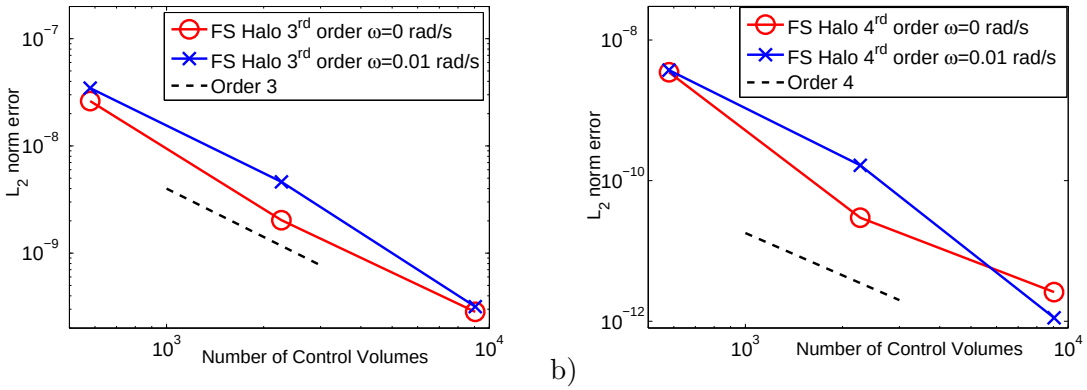
Figure 5.13: L_2 norm of the entropy error as function of the number of cells for the Ringleb flow with $\omega = 0$ rad/s. The FV-MLS scheme computations are based on three Gauss points quadrature rules. Dashed line denotes the slope of the expected formal order of accuracy.

Third-order $\omega = 0.01$ rad/s					
Method	Mesh	L_2 Entropy	Order	Conservation Error	Order
FS Intersections	580	8.10×10^{-8}	-	-	-
	2270	8.96×10^{-9}	3.23	-	-
	9044	1.14×10^{-9}	2.98	-	-
HS Intersections	580	8.76×10^{-8}	-	-	-
	2270	1.01×10^{-8}	3.16	-	-
	9044	1.27×10^{-9}	3.01	-	-
FS Halo	580	8.83×10^{-8}	-	3.46×10^{-8}	-
	2270	9.53×10^{-9}	3.36	4.62×10^{-9}	2.95
	9044	1.20×10^{-9}	2.99	3.17×10^{-10}	3.88

Table 5.3: Accuracy orders, conservation error and L_2 norm of entropy error for the Ringleb flow test case for the third-order FV-MLS scheme employing the three different approaches described in this Chapter with an angular velocity of $\omega = 0.01$ rad/s.

Fourth-order $\omega = 0.01$ rad/s					
Method	Mesh	L_2 Entropy	Order	Conservation Error	Order
FS Intersections	580	3.61×10^{-9}	-	-	-
	2270	2.78×10^{-10}	3.76	-	-
	9044	1.92×10^{-11}	3.86	-	-
HS Intersections	580	1.09×10^{-8}	-	-	-
	2270	7.25×10^{-10}	3.98	-	-
	9044	4.05×10^{-11}	4.08	-	-
FS Halo	580	3.89×10^{-9}	-	3.73×10^{-9}	-
	2270	2.85×10^{-10}	3.83	1.65×10^{-10}	4.58
	9044	1.99×10^{-11}	3.85	1.12×10^{-12}	3.89

Table 5.4: Accuracy orders, conservation error and L_2 norm of entropy error for the Ringleb test case for the fourth-order FV-MLS scheme employing the three different approaches described in this Chapter with an angular velocity of $\omega = 0.01$ rad/s.



a) b)

Figure 5.14: L_2 norm of the conservation error as function of the number of cells for the Ringleb flow for $\omega = 0$ rad/s and $\omega = 0.01$ rad/s. The FV-MLS scheme computations are based on three Gauss points quadrature rules. Dashed line denotes the slope of the expected formal order of accuracy.

5.5.2 Vortex Convection

The second test case corresponds to the unsteady isentropic vortex convection [192]. In this problem a vortex is convected with the free stream velocity (u_∞, v_∞) through the interface. This problem has an analytical smooth solution which can be subjected to obtain the order of the scheme on a unsteady case. The exact solution is defined as

$$\begin{aligned}
\frac{u(x,y,t)}{a_\infty} &= \frac{u_\infty}{a_\infty} - \frac{K}{2\pi a_\infty}\hat{y}e^{\alpha(1-r^2)/2} \\
\frac{v(x,y,t)}{a_\infty} &= \frac{v_\infty}{a_\infty} + \frac{K}{2\pi a_\infty}\hat{x}e^{\alpha(1-r^2)/2} \\
\frac{T(x,y,t)}{T_\infty} &= 1 - \frac{K^2(\gamma-1)}{8\alpha\pi^2 a_\infty^2}e^{\alpha(1-r^2)} \\
\frac{\rho(x,y,t)}{\rho_\infty} &= \left(\frac{T(x,y,t)}{T_\infty}\right)^{\frac{1}{\gamma-1}} \\
\frac{p(x,y,t)}{p_\infty} &= \left(\frac{T(x,y,t)}{T_\infty}\right)^{\frac{\gamma}{\gamma-1}}
\end{aligned}$$

where $\hat{x} = x - x_0 - u_\infty t$, $\hat{y} = y - y_0 - v_\infty t$ and $r = \sqrt{\hat{x}^2 + \hat{y}^2}$. In these simulations the chosen parameters are $\alpha = 1$, $\rho_\infty = 1$, $p_\infty = 1$, $(u_\infty, v_\infty) = (2, 2)$, $(x_0, y_0) = (-10, -10)$ and $K = 5$. With this set of parameters the vortex starts at $(x, y) = (-10, -10)$ and at $t = 5$ reaches $(x, y) = (0, 0)$.

Figure 5.15 shows a schematic description of the problem and an unstructured mesh of 3884 quadrilateral elements employed in the error convergence study.

First, we obtain the L_2 norm of the variables error, conservation error and corresponding orders of accuracy for the case with $\omega = 0$ rad/s and non-conformal meshes. The results are presented in Table 5.5. Formal third order is reached for the variables error for the three approaches presented in this Chapter. All methods give similar results in terms of accuracy.

In Table 5.6 the results with $\omega = 1.0$ rad/s are shown. It is observed that with rotation the accuracy decreases but the formal order of convergence is not affected.

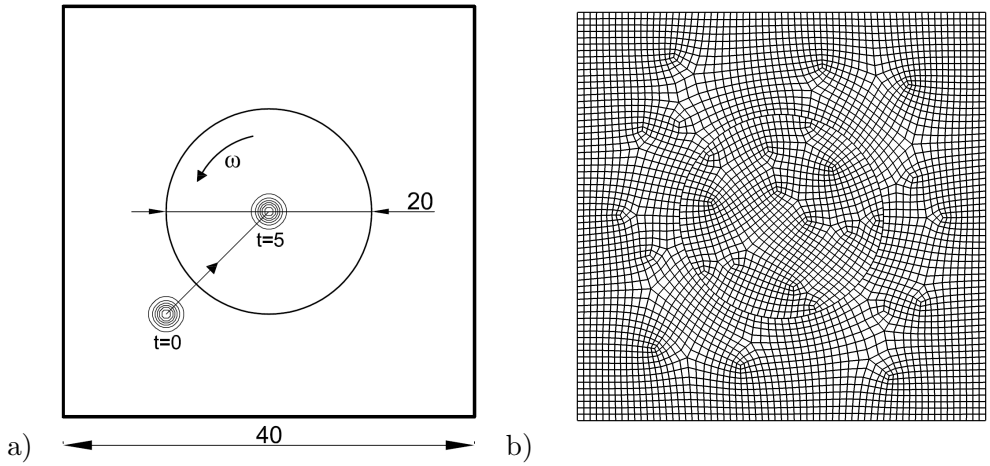


Figure 5.15: a) Geometry description of Two Dimensional Vortex Convection problem. b) Unstructured mesh of 3884 quadrilateral elements.

Third-order $\omega = 0$ rad/s					
Method	Mesh	L_2	Order	Conservation Error	Order
FS Intersections	3884	6.03×10^{-2}	-	-	-
	16407	9.34×10^{-3}	2.59	-	-
	52136	1.07×10^{-3}	3.74	-	-
HS Intersections	3884	6.27×10^{-2}	-	-	-
	16407	9.23×10^{-3}	2.66	-	-
	52136	1.03×10^{-3}	3.80	-	-
FS Halo	3884	6.49×10^{-2}	-	1.07×10^{-1}	-
	16407	1.08×10^{-2}	2.49	1.34×10^{-2}	2.88
	52136	1.11×10^{-3}	3.93	1.14×10^{-3}	4.27

Table 5.5: Accuracy orders, conservation error and L_2 norm of variables error for the two dimensional vortex convection for the third-order FV-MLS scheme employing the three different approaches described in this Chapter with an angular velocity of $\omega = 0.0$ rad/s.

Third-order $\omega = 1.0$ rad/s					
Method	Mesh	L_2	Order	Conservation Error	Order
FS Intersections	3884	6.26×10^{-2}	-	-	-
	16407	1.26×10^{-2}	2.23	-	-
	52136	1.95×10^{-3}	3.22	-	-
HS Intersections	3884	6.10×10^{-2}	-	-	-
	16407	1.32×10^{-2}	2.13	-	-
	52136	1.94×10^{-3}	3.31	-	-
FS Halo	3884	6.58×10^{-2}	-	9.74×10^{-2}	-
	16407	1.31×10^{-2}	2.24	1.52×10^{-2}	2.58
	52136	1.92×10^{-3}	3.32	1.31×10^{-3}	4.24

Table 5.6: Accuracy orders, conservation error and L_2 norm of variables error for the two dimensional vortex convection for the third-order FV-MLS scheme employing the three different approaches described in this Chapter with an angular velocity of $\omega = 1.0$ rad/s.

5.5.3 Supersonic Flow over a cylinder

In this section the obtained results of the numerical method applied to the simulation of the inviscid supersonic flow over a cylinder are presented. The aim of this problem is to analyze the performance of the Interface Halo cell Sliding Mesh approach with supersonic flows and to study the suitability of the halo formulation to deal with shocks through the interface in a multidimensional problem. The problem set-up is presented in Figure 5.16.

The computational domain is discretized with a non-conformal mesh of 7200 quadrilateral elements. Figure 5.17 shows the mesh and a close view of the interface.

The freestream Mach number is $M_\infty = 3.0$. Following [151], the shock detector based on MLS, described in section 3.4.2, and the limiter of Van Albada [210], described in section 3.4.1, were used.

The benchmark parameters for this test case are the pressure coefficient C_p, the normalized stagnation pressure p_0 and the stand-off distance normalized by the diameter of the cylinder. The stand-off distance is defined in Figure 5.18.

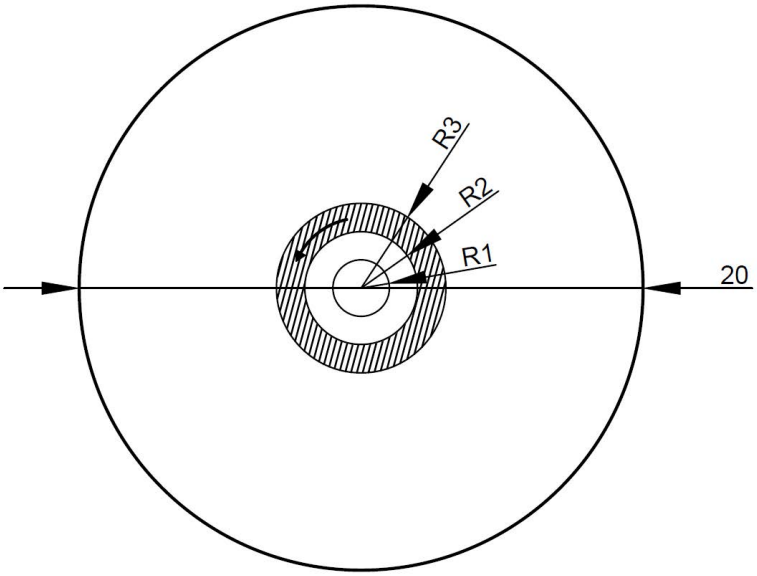


Figure 5.16: Geometry description of the Supersonic Flow over a cylinder of radius $R1 = 1$. The shaded ring, with inner radius $R2 = 2$ and outer radius of $R3 = 3$, denotes the rotational zone.

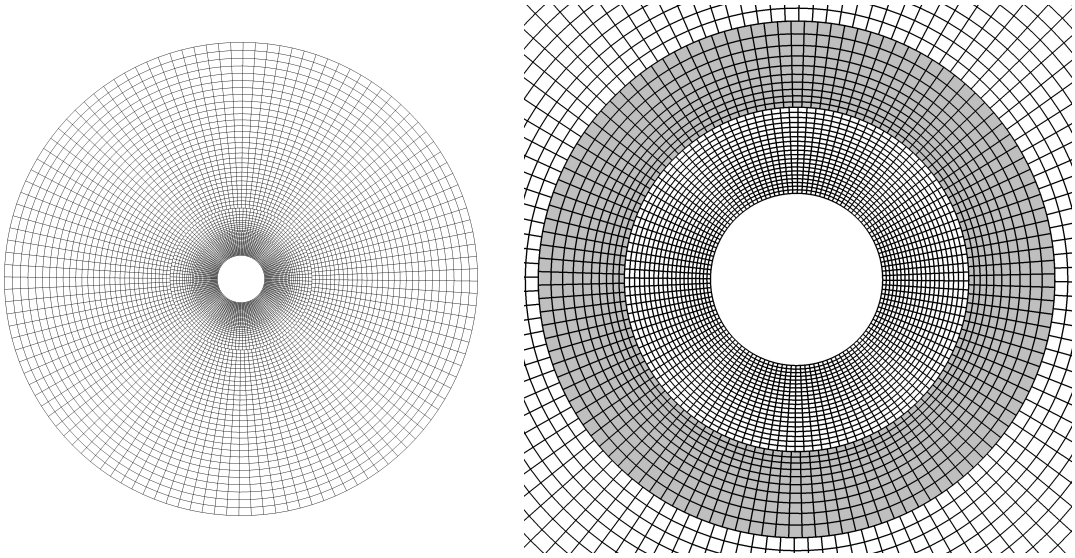


Figure 5.17: Structured non-conformal mesh of 7200 quadrilateral elements and closer view to the cylinder and the interfaces.

The numerical results obtained for Mach number isolines with sliding mesh FS intersections and FS halo cell are plotted in Figure 5.19 for a rotational velocity of 1000

rpm. In order to compare the results we have computed the case on a conformal single mesh with no interface. Almost no differences can be observed between the different solutions.

In Figure 5.20 the pressure coefficient around the cylinder is compared for the different approaches. Again no differences are observed. In Table 5.7, the obtained results are shown for the normalized stagnation pressure p_0 and the stand-off distance normalized by the diameter of the cylinder. The sliding-mesh computations give the same results. These values have an excellent agreement with the ones obtained for a single grid and the reference solution [119].

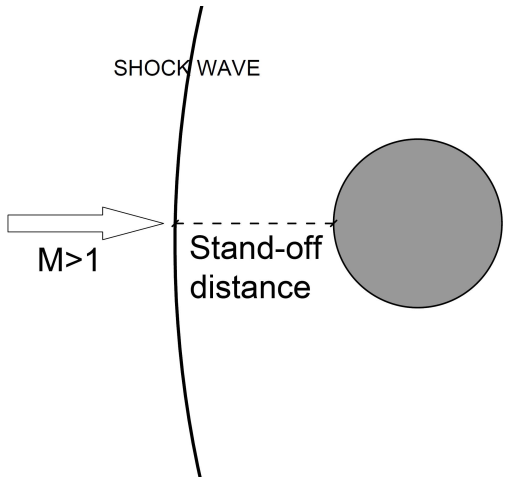


Figure 5.18: Schematic representation of the stand-off distance.

Method	$p_0/(p_0)_\infty$	Stand-off distance/D
Single mesh	0.327	0.405
Sliding Mesh FS Halo 0 rpm	0.324	0.407
Sliding Mesh FS Halo 1000 rpm	0.324	0.408
Sliding Mesh FS Intersections 1000 rpm	0.324	0.408
Reference solution [119]	0.328	-

Table 5.7: Comparison of the normalized stagnation pressure p_0 and the stand-off distance obtained using a single mesh and the three sliding mesh methods (present computations were performed using a 3rd order FV-MLS scheme)

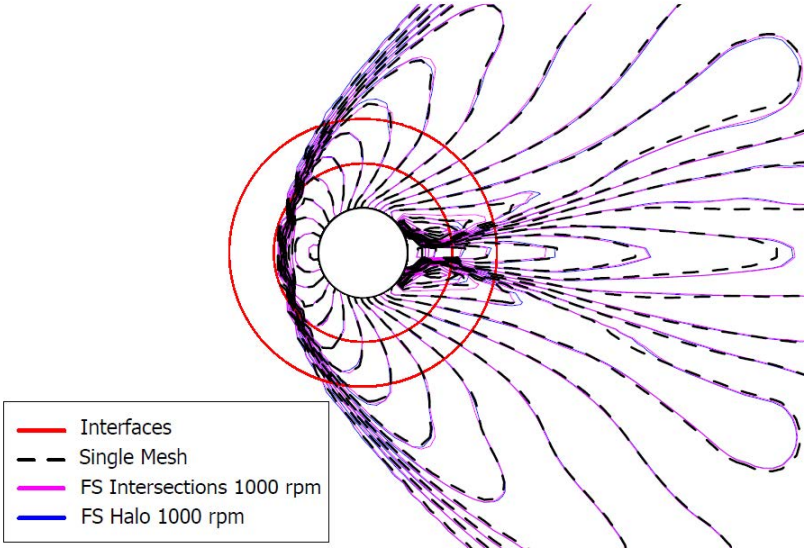


Figure 5.19: Comparison of Mach field for the supersonic flow over a cylinder. The solutions are obtained using third-order FV-MLS scheme. Dashed line denotes the solution obtained with a single mesh, purple line is obtained with FS Intersections approach and blue line refers to the solution obtained when the FS Halo approach is employed. The interfaces are highlighted in red.

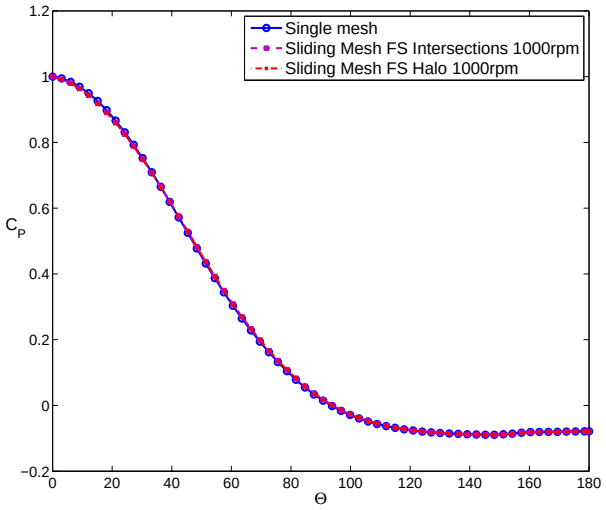


Figure 5.20: Comparison of the C_p distribution the supersonic flow over a cylinder. The solutions are obtained using third-order FV-MLS scheme.

5.5.4 Cross-flow turbines

There are main two types of vertical axis cross-flow turbines based on their primary source of propulsion: drag-based turbines and lift-based turbines. The first type is based on the drag force and is often called the Savonius-type turbine (5.21). The fundamental behind this kind of turbine is the difference of the drag coefficient between the two blades, which gives the torque of the turbine.

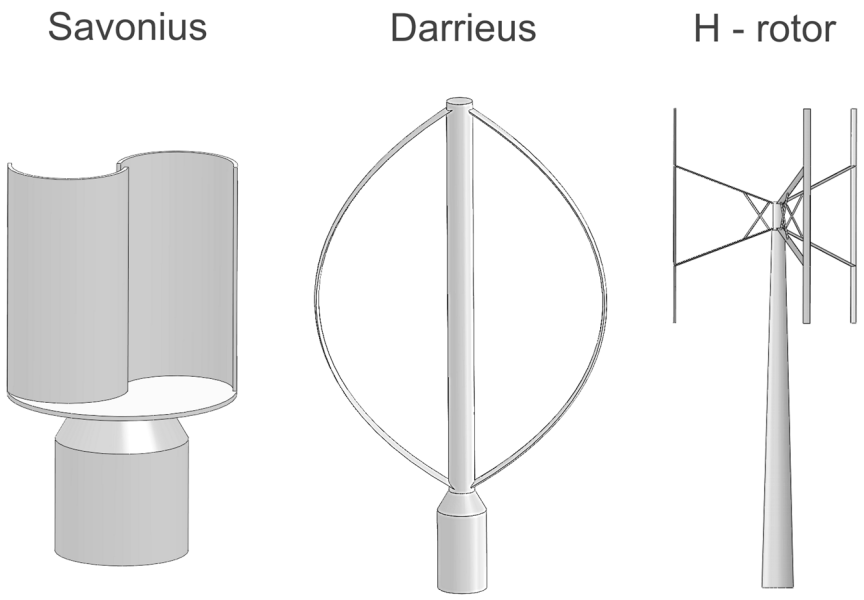


Figure 5.21: Different types of cross-flow turbines. From [67]

The lift-based turbines are the most common turbine. In this category are englobed the Darrieus-type and the H-rotor vertical axis turbines.

In order to demonstrate the capabilities of the higher-order sliding-mesh interfaces to deal with viscous flows on complex configurations, the unsteady flow is computed through two cross-flow turbines. As seen in Table 1.2, the fluid can be considered incompressible. The problem setup considered in the present work was defined by Ferrer et al. in [58]. Such cross-flow turbine configuration leads to the occurrence of complex flow phenomena, such as blade-vortex interactions.

In this test case the Interface-halo-cell approach has been used.

In order to compare the results with the reference solution [57], the force vector is computed in cartesian coordinates as

$$\boldsymbol{f} = \left\{ \begin{array}{c} f_x \\ f_y \end{array} \right\} = \oint (p\boldsymbol{n} - \nu(\nabla \boldsymbol{U} \cdot \boldsymbol{n}))d\Gamma \tag{5.15}$$

where $\boldsymbol{n}$ is the outward pointing normal at the airfoil. Once the forces are obtained in Cartesian coordinates, they can be expressed on a normal-tangential frame of reference as

$$\begin{aligned} f_N &= f_y \cos\theta - f_x \sin\theta \\ f_T &= -f_x \cos\theta - f_y \sin\theta \end{aligned} \tag{5.16}$$

where θ is the angular location. Figure 5.22 shows a diagram of the angles and velocities on a single blade. The forces can be expressed on a drag-lift frame of reference as

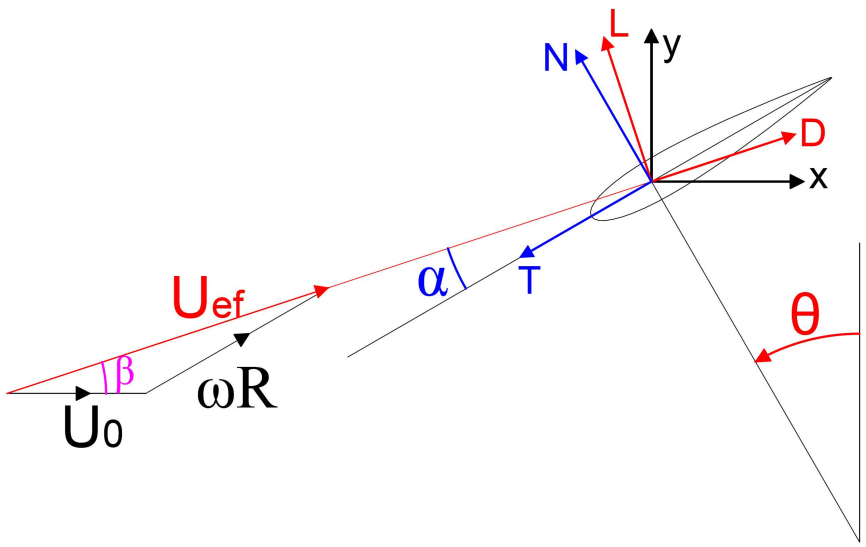


Figure 5.22: Diagram of velocities, angles and frames of references on a single blade. The cartesian frame of reference is represented by (x, y), the drag-lift fram of reference by (D, L) and the normal-tangential frame by (N, T)

$$\begin{aligned} f_D &= f_x \cos\beta + f_y \sin\beta \\ f_L &= f_y \cos\beta - f_x \sin\beta \end{aligned} \tag{5.17}$$

where β is defined as $\beta = \theta - \alpha$. The angle of attack α corresponds to the angle between the chord line and the direction of the relative flow direction. Since the turbine rotates α varies throughout the rotation cycle as shown in Figure 5.22.

The angle of attack can be expressed as function of the tip speed ratio and the angular location θ as [131]:

$$\alpha = \tan^{-1}\left(\frac{\sin\theta}{\cos\theta + \lambda}\right) \tag{5.18}$$

The Reynolds number is defined as $Re = \frac{U_0 c}{\nu}$. The tip speed ratio is defined as $\lambda = \omega R/U_0$

In Figure 5.23, the variation of the angle of attack of a blade (α) for a rotation cycle with a tip speed ratio of $\lambda = 2$ is plotted.

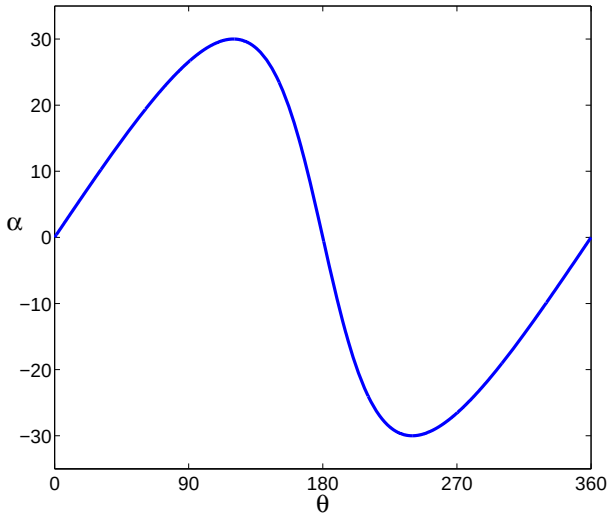


Figure 5.23: Representation of the angle of attack (α) as function of the angular location (θ) for tip speed ratio of $\lambda = 2$.

Single-bladed cross-flow turbine

The first test case is the study of the flow over a single-bladed cross-flow turbine. The basic configuration of this problem it is shown in Figure 5.24. The turbine is formed with a single NACA 0015 airfoil with a chord $c = 1$ located at a radial distance $R = 2c$ from the center of rotation.

A close view of the computational grid near the rotating grid is presented in Figure 5.25.

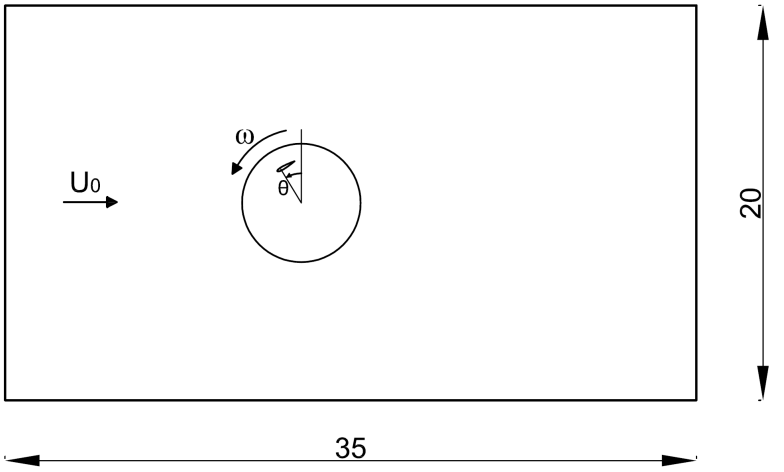


Figure 5.24: Schematic description of the one-bladed cross-flow turbine

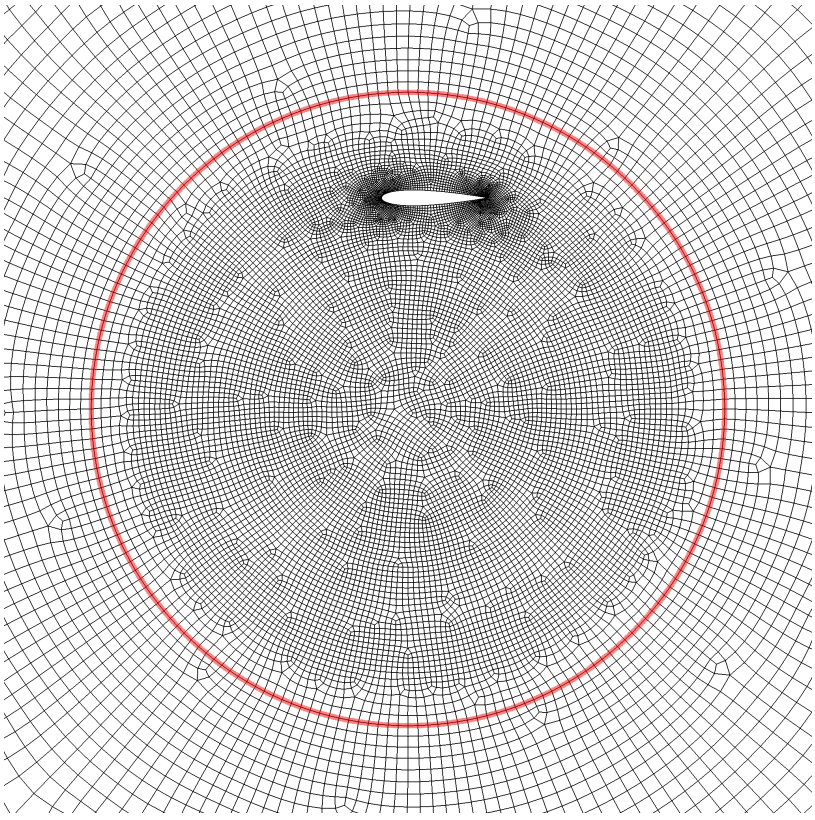


Figure 5.25: Close view of the unstructured mesh around the single-bladed cross flow turbine. The red line denotes the interface

In this example, three different speed tip ratio ($\lambda = 1, 2, 5$) are studied. The initial conditions are described in Table 5.8

Free-stream velocity U_0	Rotational Speed ω	Tip Speed Ratio λ
0.2	0.5	5
0.5	0.5	2
1.0	0.5	1

Table 5.8: Initial Conditions for the One-bladed cross flow turbine

Figure 5.26 compare the normalized tangential and normal forces, defined in equation 5.17, against the angular rotation θ.

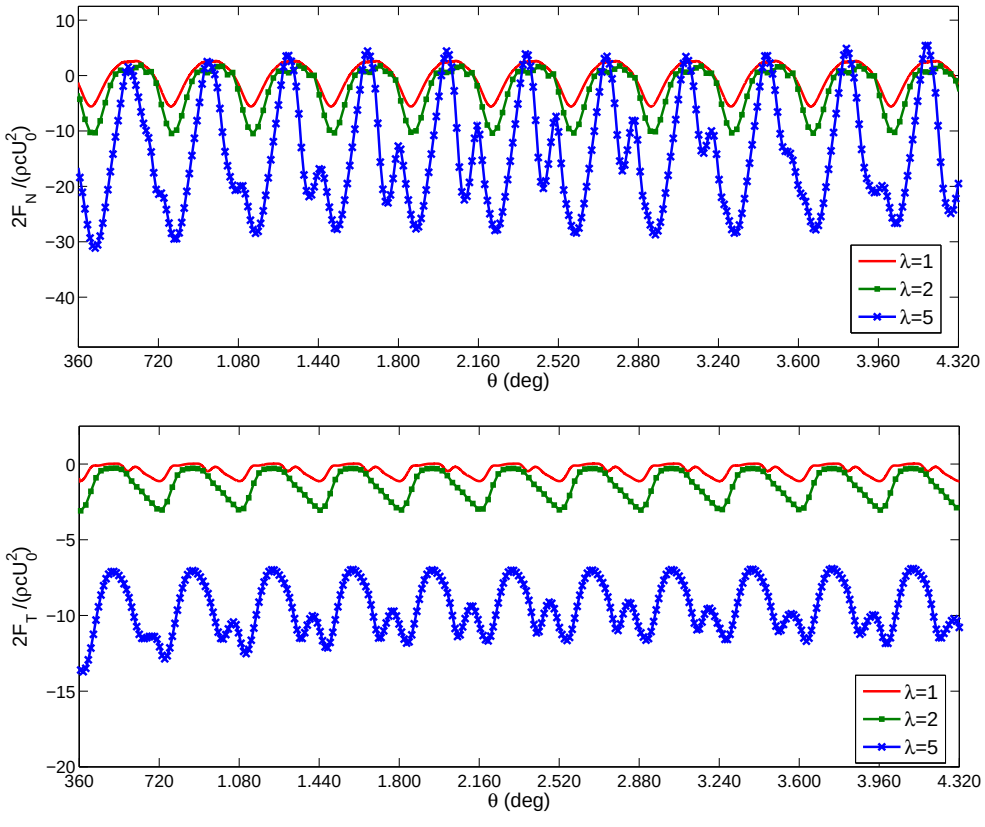


Figure 5.26: a) Normalized normal force and b) normalized tangential force against azimuth for $\lambda = 1, 2, 5$ for the one-bladed cross-flow turbine.

A good agreement is achieved with the results obtained by Ferrer [57] using a 3^{rd} order Discontinuous Galerkin method. The velocity contours, pressure and vorticity fields near the NACA after six rotation cycles and $\lambda = 5$ are shown in Figures 5.27 and 5.28. A smooth solution is obtained across the interface. No numerical artifacts are observed.

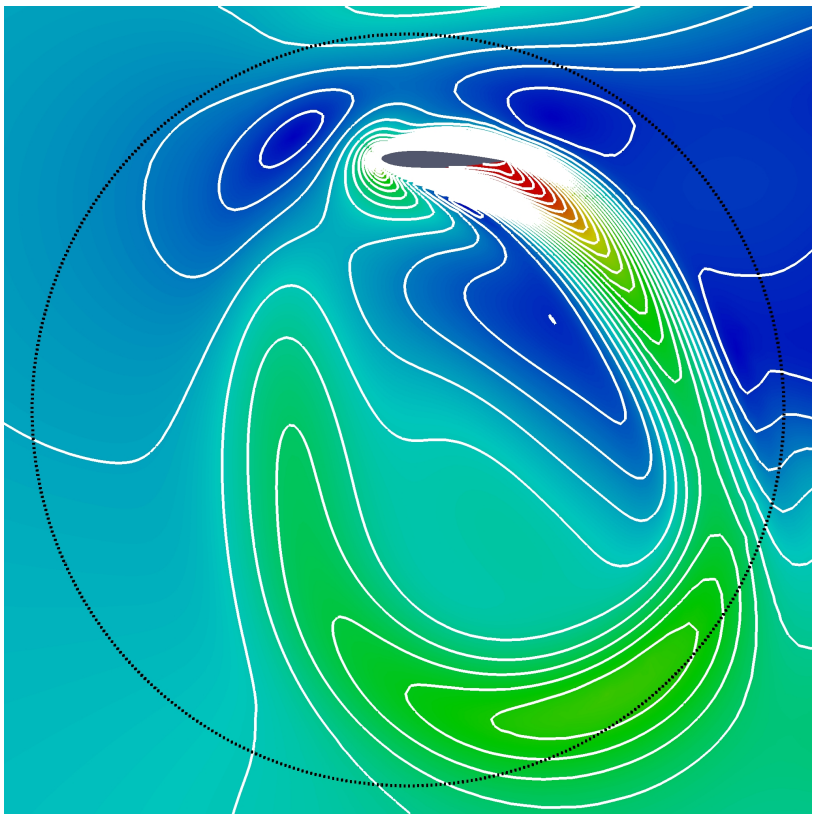


Figure 5.27: Velocity contours around the one bladed cross flow turbine. The solution is obtained with a 3^{rd} FV-MLS method after six rotation cycles ($\theta = 1800$ deg) with $\lambda = 5$

Three bladed cross-flow turbine

The basic configuration of the problem is shown in Figure 5.29. The turbine is formed with three NACA0015 airfoil of chord $c = 1$. The blades are disposed with a relative angle of 120 degrees as seen in Figure 5.30.

The free-stream velocity is $U_0 = 0.5$. The tip speed ratio is equal to $\lambda = 2$. The Reynolds number is $Re = 50$ and the rotational velocity $\omega = 0.5$. The computational domain is discretized with an unstructured mesh of 26234 quadrilateral elements. Note that with this number of elements, the spatial resolution of the mesh is similar to the one in [58]. A close view of the mesh near the rotating grid is presented in Figure 5.30.

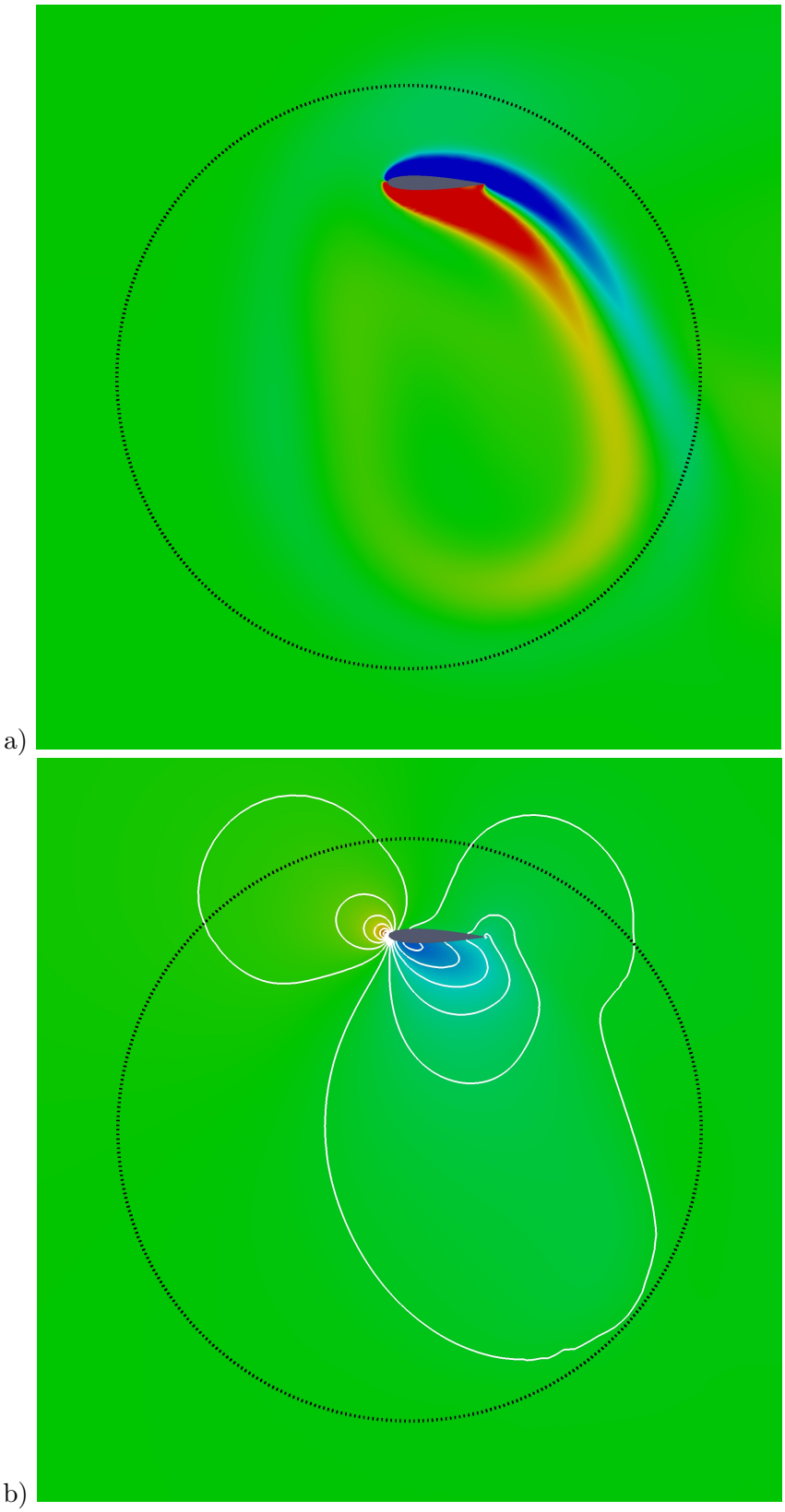


Figure 5.28: Flow around the one bladed cross-flow turbine, vorticity field (a) and pressure field (b).The solution is obtained with a 3^{rd} FV-MLS method after six rotation cycles ($\theta = 1800$ deg) with $\lambda = 5$

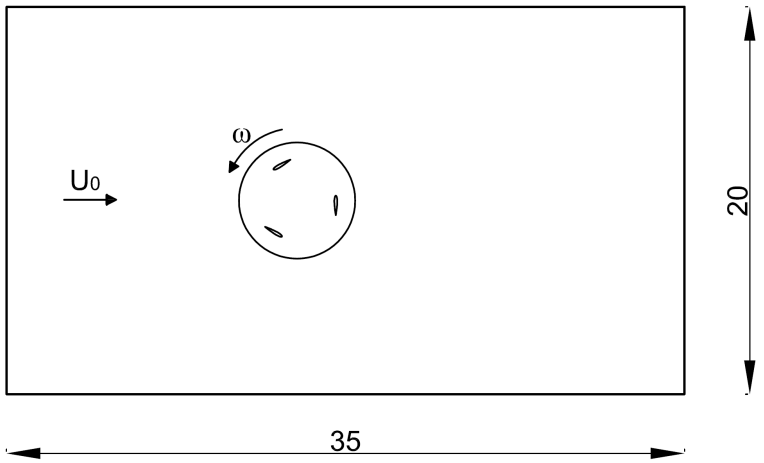


Figure 5.29: Schematic description of the three-bladed cross-flow turbine.

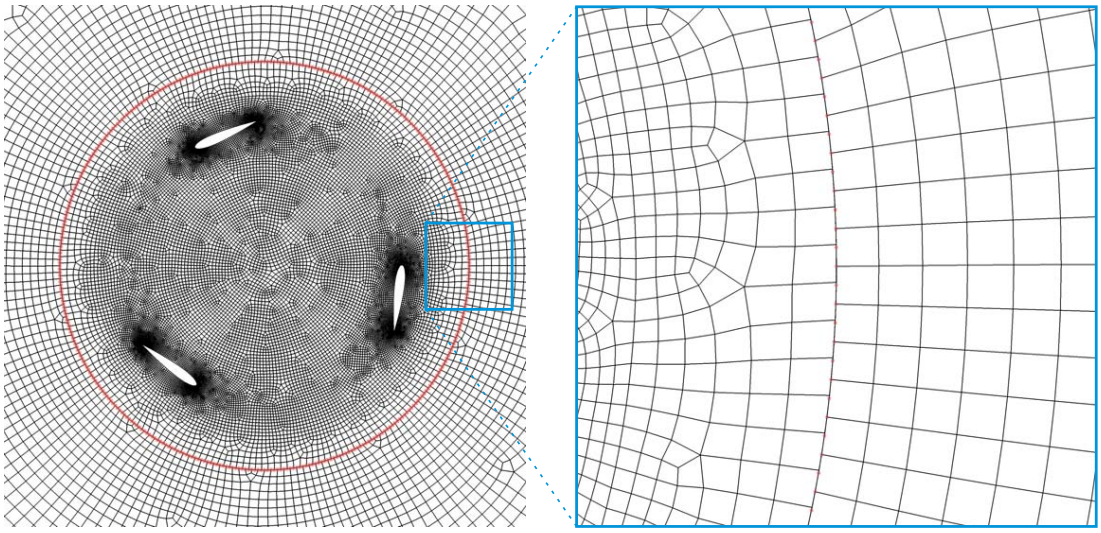


Figure 5.30: Close view of the unstructured mesh around the cross flow turbine at time $t = 5.0$. Note the non-conformal mesh at the interface. The red line denotes the interface

The solution obtained at time $t = 5$ is plotted in terms of the velocity field and contours in Figure 5.32, pressure field and contours Figure 5.31 a) and vorticity field 5.31 b). Note that there is no any visible numerical artifacts at the interface.

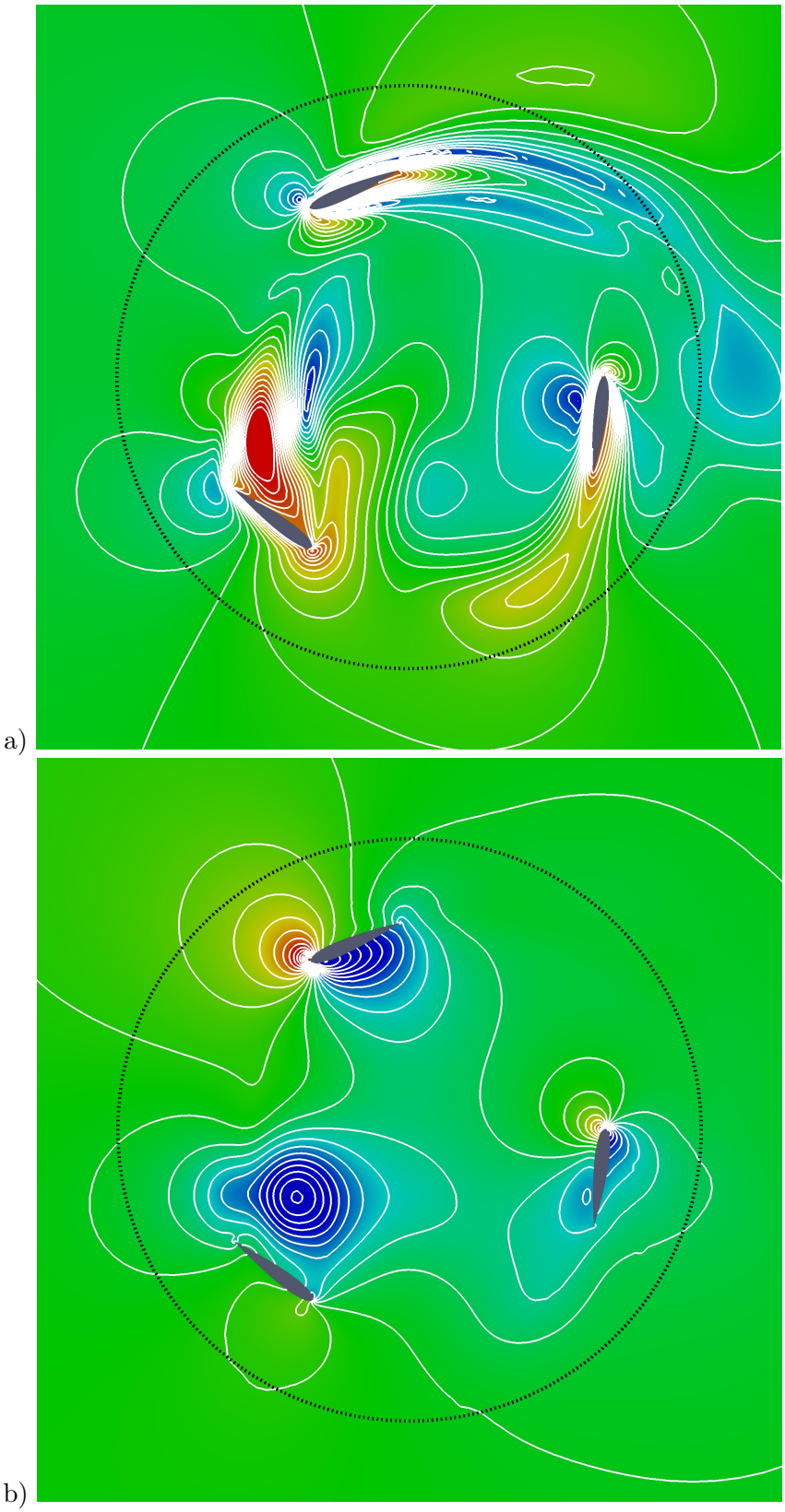


Figure 5.31: Flow around the three bladed cross-flow turbine, velocity field (a) and pressure field (b). The solution is obtained with a third-order FV-MLS method at $t = 5.0$.

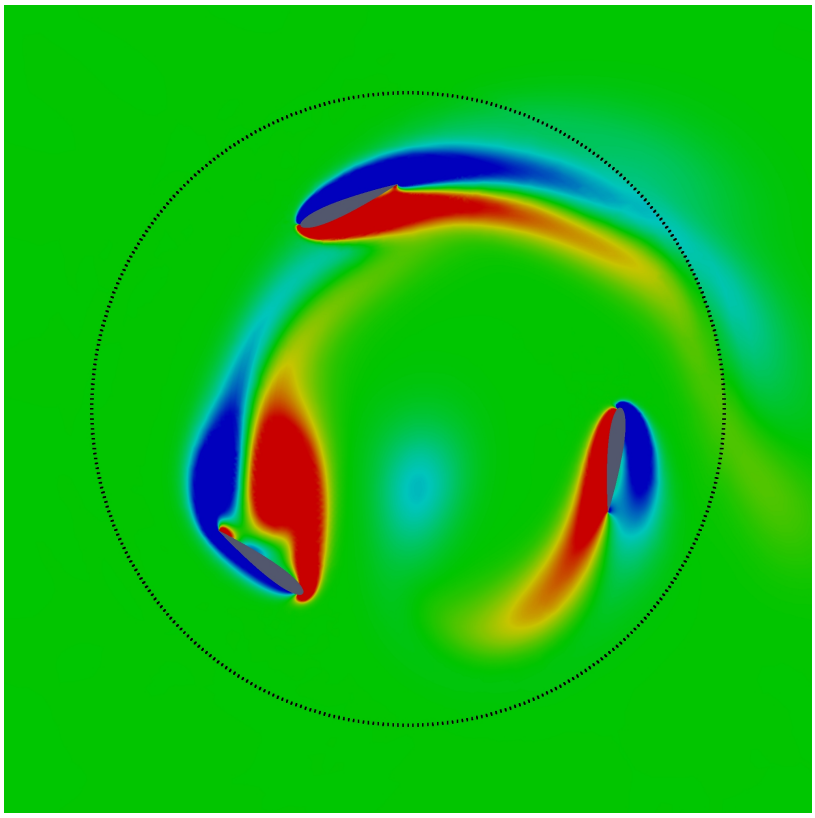


Figure 5.32: Vorticity field and contours around the three bladed cross-flow turbine. The solution is obtained with a third-order FV-MLS method at $t = 5.0$.

Figure 5.33 compare the normalized tangential and normal forces against the angular rotation θ for one of the blades of the turbine. Results are in good agreement with those obtained by Ferrer [57] using a third-order Discontinuous Galerkin method.

Figure 5.34 plots the normalized drag and lift forces for one of the blades of the three-bladed cross-flow turbine against the angular location θ.

The numerical solutions obtained for different angular locations are plotted in Figure 5.35 and 5.36 in terms of the velocity magnitude and in Figures 5.37 and 5.38 in terms of the pressure field. In Figure 5.39 and 5.40 the vorticity field is plotted for different angular locations θ. Again, no numerical artifacts near the interface are observed and the velocity and pressure isolines are perfectly smooth across the non conformal grids.

High-accurate numerical methods can help engineers to a better understanding of the interaction between flow and turbine, which can lead to new designs and an improvement of the efficiency of existing devices. For example, in a cross-flow turbine, different configurations can be tested in order to enhance the efficiency of the device.

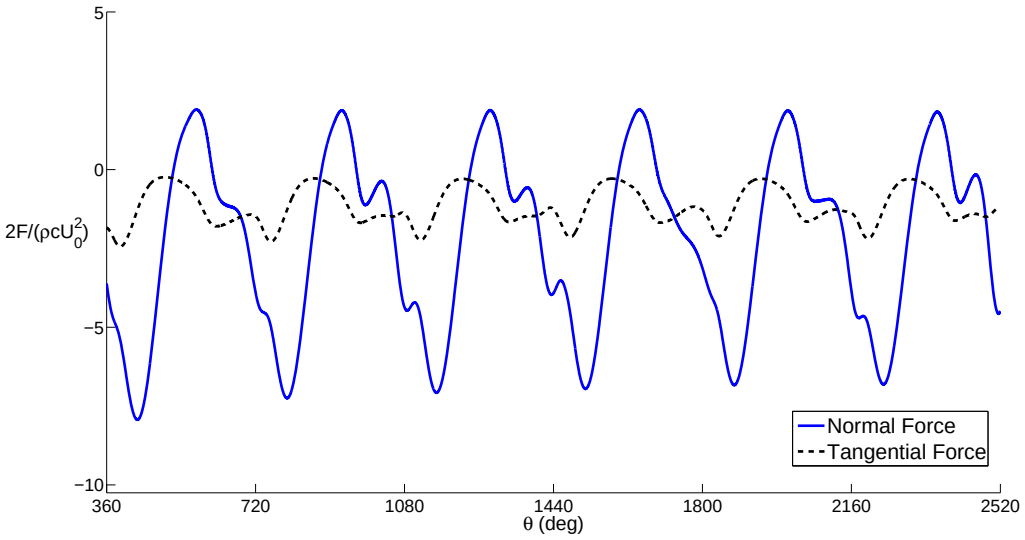


Figure 5.33: Normalized tangential and normal forces against azimuth for one of the blades of the three-bladed cross-flow turbine.

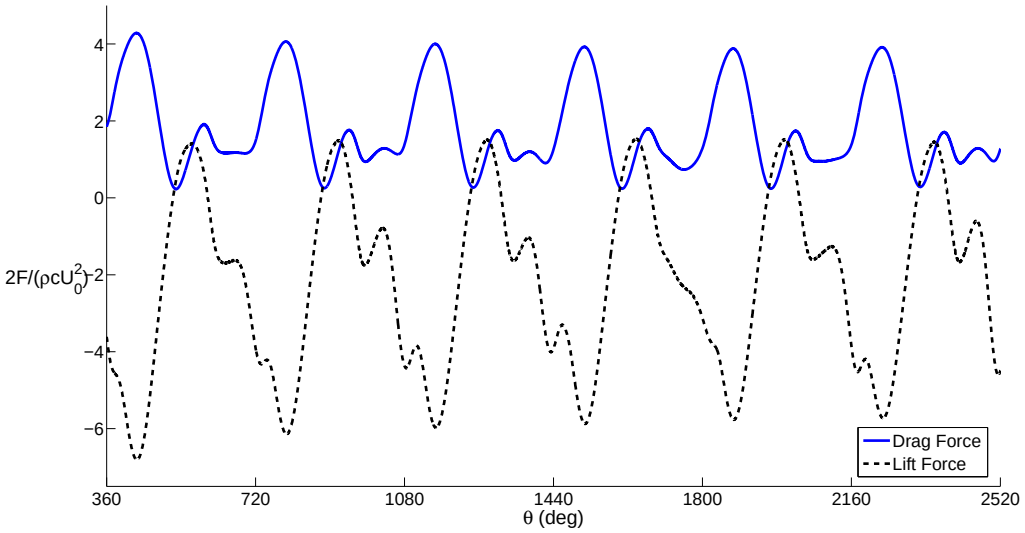


Figure 5.34: Normalized drag and lift forces against azimuth for one of the blades of the three-bladed cross-flow turbine.

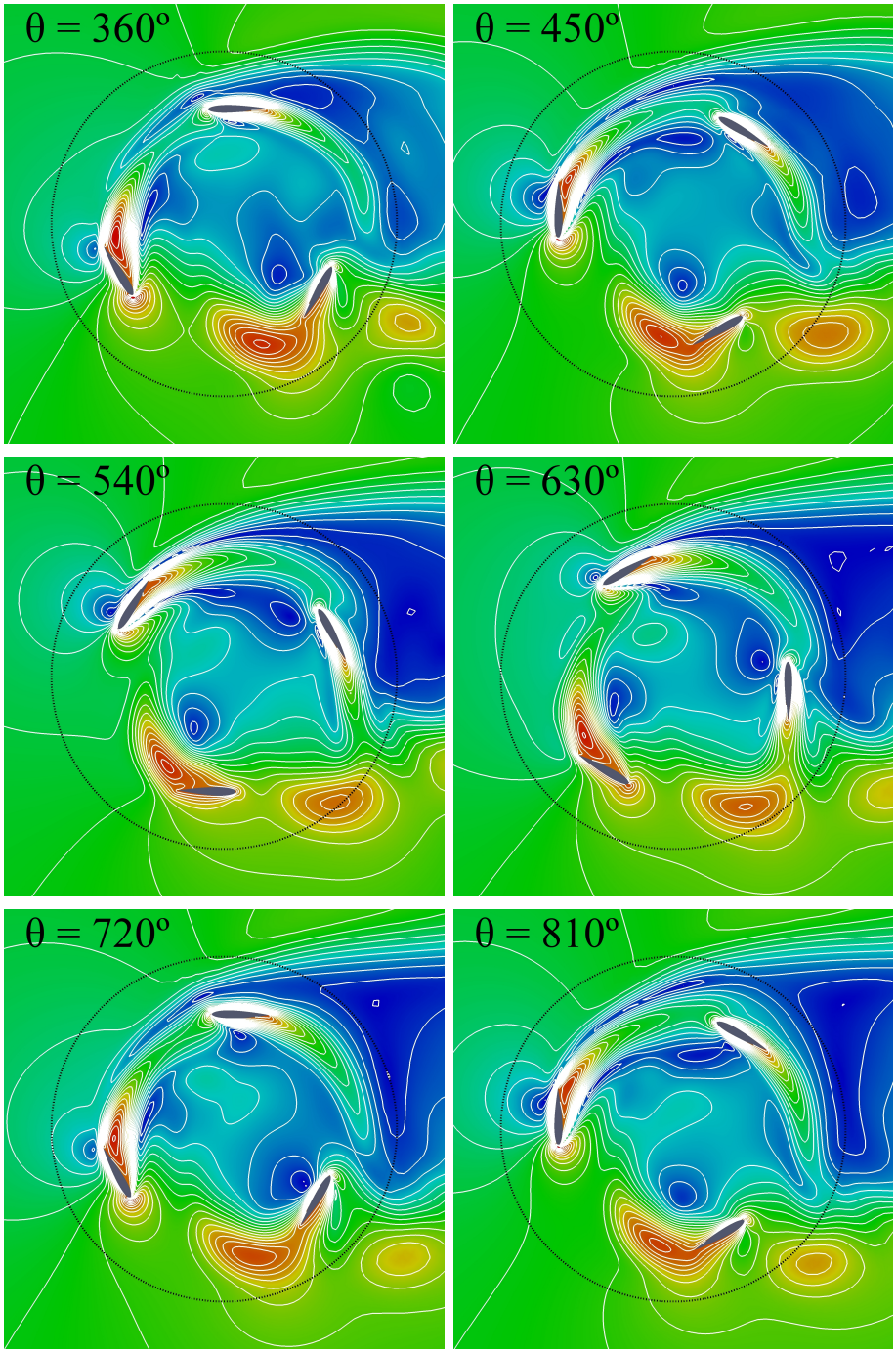


Figure 5.35: Velocity field and contours around the three bladed cross-flow turbine. The solutions are obtained using a third-order FV-MLS method. The solution is plotted for the following angular locations: $\theta = 360, 450, 540, 630, 720, 810$ deg. Note that no numerical artifacts near the interface are observed.

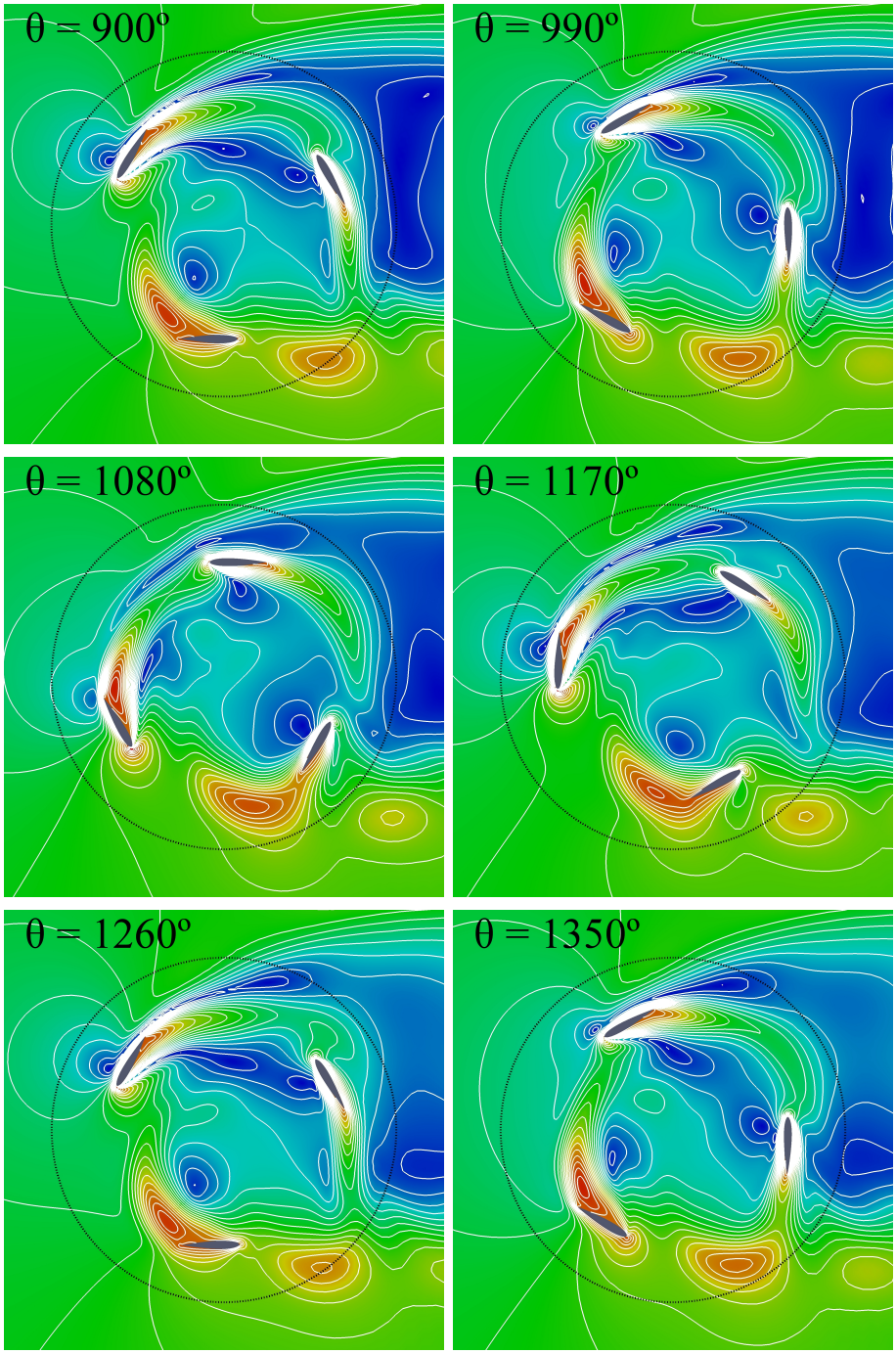


Figure 5.36: Velocity field and contours around the three bladed cross-flow turbine. The solutions are obtained using a third-order FV-MLS method. The solution is plotted for the following angular locations: $\theta = 900, 990, 1080, 1170, 1260, 1350$ deg. Note that no numerical artifacts near the interface are observed.

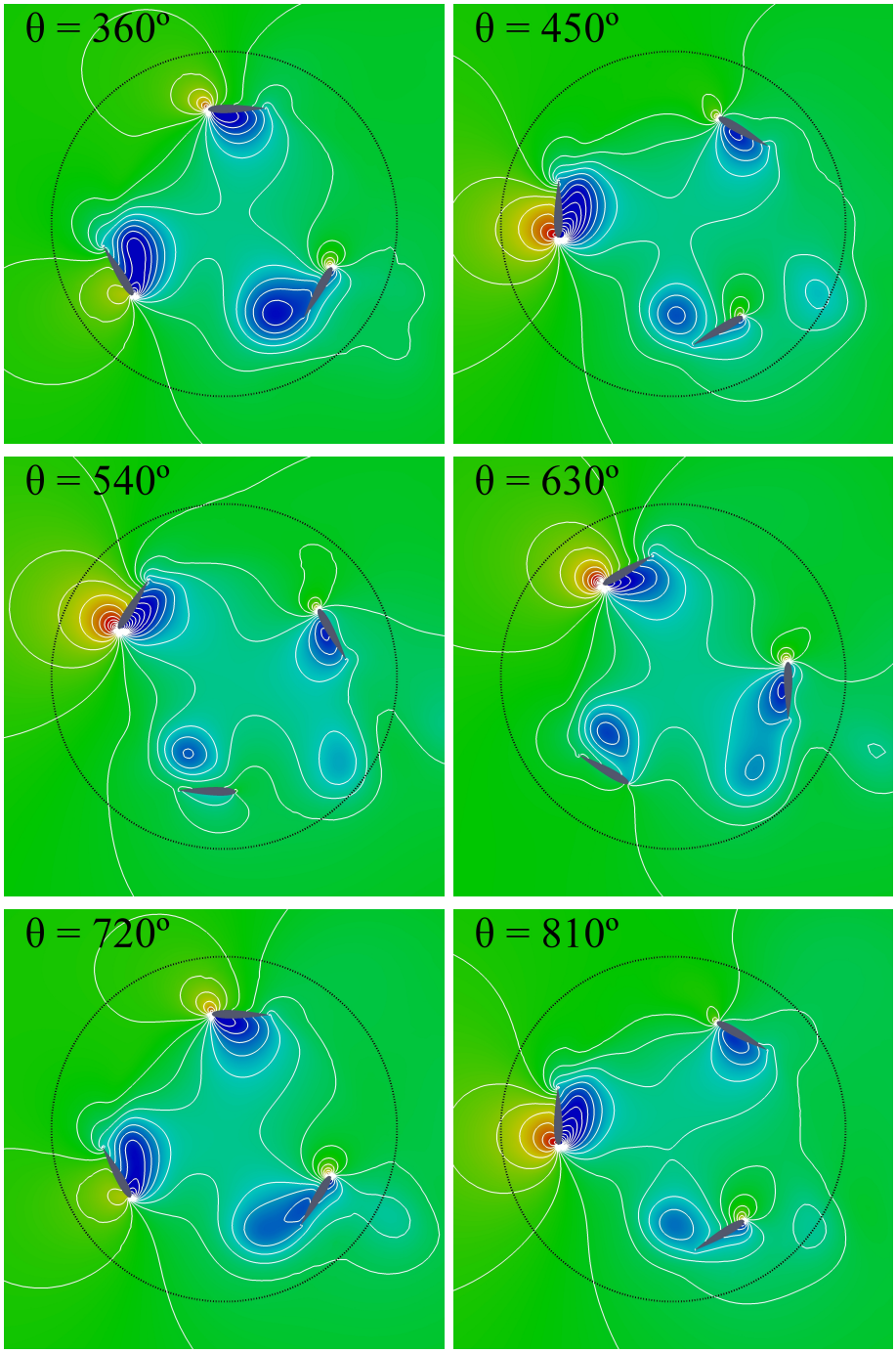


Figure 5.37: Pressure field and contours around the three bladed cross-flow turbine. The solutions are obtained using a third-order FV-MLS method. The solution is plotted for the following angular locations: $\theta = 360, 450, 540, 630, 720, 810$ deg. Note that no numerical artifacts near the interface are observed.

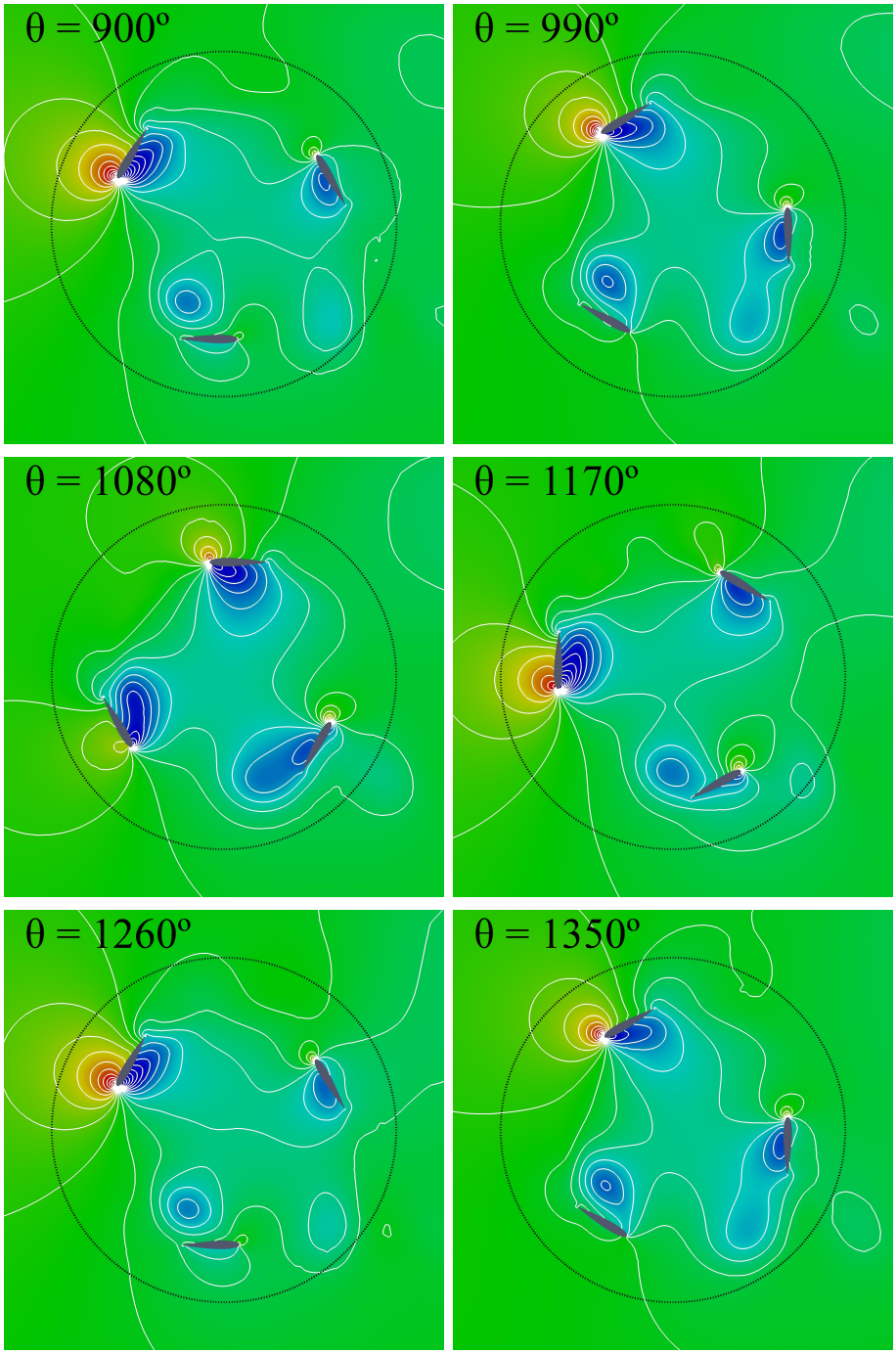


Figure 5.38: Pressure field and contours around the three bladed cross-flow turbine. The solutions are obtained using a third-order FV-MLS method. The solution is plotted for the following angular locations: $\theta = 900, 990, 1080, 1170, 1260, 1350$ deg. Note that no numerical artifacts near the interface are observed.

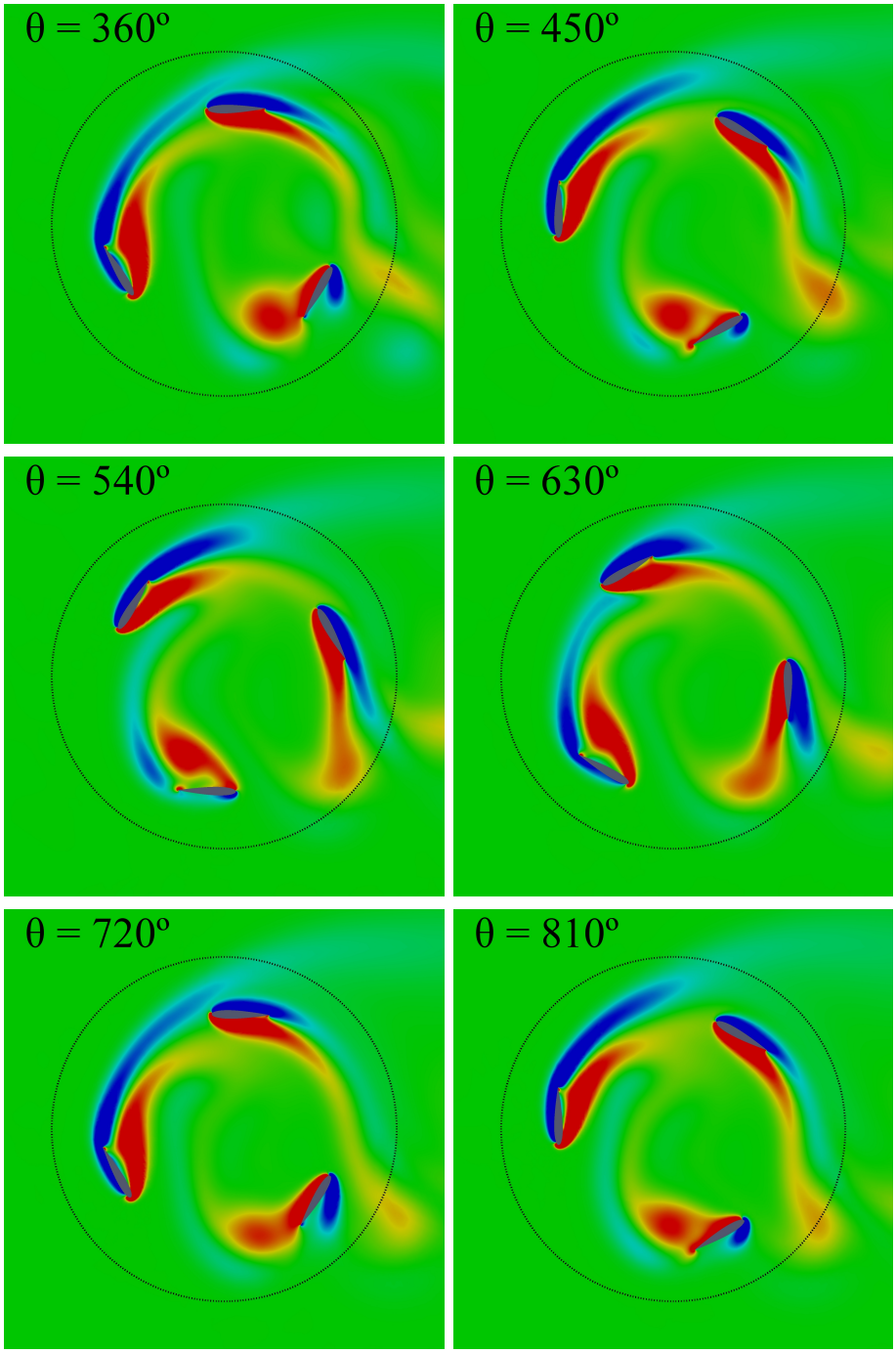


Figure 5.39: Vorticity field around the three bladed cross-flow turbine. The solutions are obtained with a third-order FV-MLS method for the following angular locations: $\theta = 360, 450, 540, 630, 720, 810$ deg. Note that no numerical artifacts near the interface are observed.

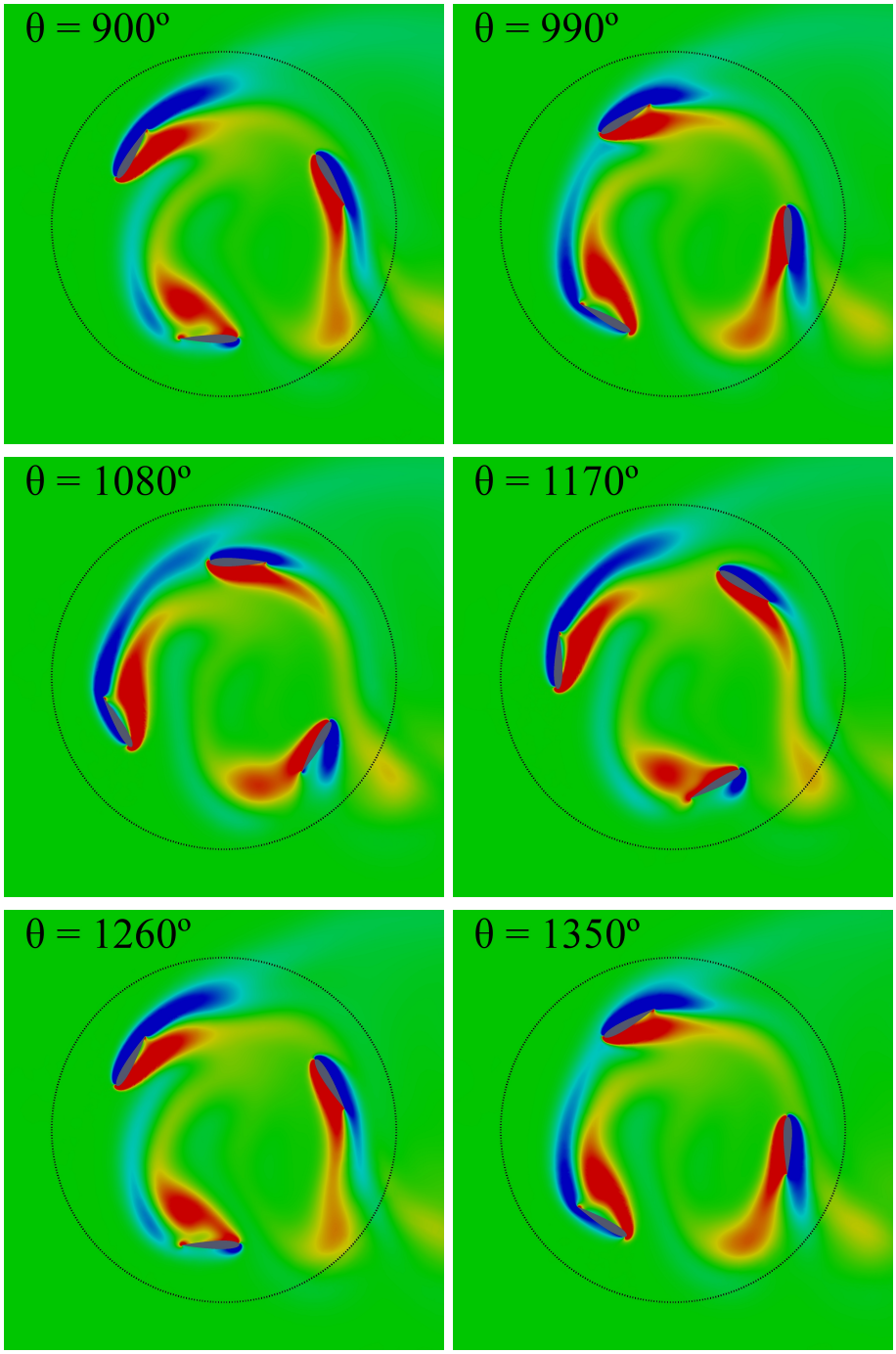


Figure 5.40: Vorticity field around the three bladed cross-flow turbine. The solutions are obtained with a third-order FV-MLS method for the following angular locations: $\theta = 900, 990, 1080, 1170, 1260, 1350$ deg. Note that no numerical artifacts near the interface are observed.

5.6 Conclusions

In order to numerically simulate the flow on a turbomachine, the relative motion of the blades needs to be taken into account. In order to preserve the accuracy of the numerical scheme, the simulation of rotating geometries needs to be of, at least, the same order than the numerical scheme. This is a crucial point in the development of high-order methods for the simulations of turbomachines. However, most of the sliding mesh methods proposed in the literature belong to the family of second-order interpolation schemes at most. Therefore they can not be used in conjunction with higher-order numerical schemes without depreciating the overall accuracy of the numerical methods.

In this Chapter, high-order-preserving sliding-mesh methods based on Moving Least Squares approximants are presented.

Two distinct approaches were considered, namely the intersection-based and the *halo cell* sliding mesh methods. The latter is more flexible since it avoids the computation of intersections. As a drawback, it is not possible to assure mass conservation. However numerical results show that conservation errors are within the order of magnitude of the error committed on the variables and the rate of convergence is not affected.

The accuracy and robustness of the new methodology has been investigated using various numerical test for both inviscid compressible and viscous incompressible flows. Numerical results have shown that the proposed high-resolution sliding-mesh methods are able to preserve the formal order of accuracy of the high-order spatial discretization scheme.

The *halo cell* sliding mesh method has been applied to the computation of a three-bladed cross flow turbine, and the results are in good agreement with the results obtained with other high-order sliding mesh approaches based on intersections.

Chapter 6

Conclusions and further researh

"The important thing is not to stop questioning. Curiosity has its own reason for existing."

- Albert Einstein.

This thesis has developed and analyzed a high-order finite volume method based on the use of Moving Least Squares, the FV-MLS, for the numerical resolution of the Navier-Stokes equations for all-speed and incompressible flows in complex geometries. In addition, in order to simulate accurately the flow around rotating geometries, such as the blades of a turbine, new high-order sliding mesh methods are proposed. One of the main application these new numerical techniques is the computation of flows in turbomachines.

- In Chapter 3 a high-order finite volume method for the resolution of the compressible Navier-Stokes equations on complex geometries for all-speed flows has been developed.

  The main ingredients of the formulation rely on the MLS-based finite volume formulation presented in Chapter 2, a low-Mach fix and a slope limiting strategy coupled with an MLS-based shock sensor. It is demonstrated that the use of a high-order discretization scheme alleviates the accuracy problem at low Mach numbers. However, the accuracy of the solution presents a grid dependency with the Mach numbers . In this thesis, the accuracy problem was thoroughly investigated for both the fluxes of Roe and Rusanov. In order to circumvent this problem, it is proposed the use of high-order schemes along with low-Mach fixes, successfully preserving the accuracy of the solution at low Mach numbers. The high-order density-based finite volume methodology developed for all-speed flows must include a limiting mechanism in order to be Total Variation Diminishing (TVD) for transonic and supersonic cases. However, it is shown that even at low Mach numbers as $M = 10^{-3}$ the limiter was activated.

The activation of a non-differentiable limiter in low Mach regions of the flow destroys the accuracy of the low-Mach fix. In order to keep the accuracy of the high-order scheme, an MLS-based shock wave sensor is applied. This sensor prevents unnecessarily activation of the slope limiter, thus avoiding the presence of spurious pressure oscillations in low-Mach regions. The accuracy of the scheme has been proved for low Mach numbers and several steady and unsteady examples were presented.

From a practical point of view, density-based solvers require small time steps to be used in flows where the Mach number is low in all the computational domain, due the stiffness of the equations at the incompressibility limit.

- In Chapter 4 a high-order pressure-based finite volume scheme for the resolution of the incompressible Navier-Stokes equations on complex geometries has been developed.

  The new formulation is based on the Moving Least Squares (MLS) approximations. Incompressibility condition is imposed iteratively each time step using the SIMPLE algorithm. The Moving Least Squares approach is employed in order to obtain higher-order approximations and derivatives that allows us to obtain a higher-order numerical method on unstructured collocated grids. The Momentum Interpolation Method (MIM) is used to avoid numerical instabilities. The usual linear formulation of MIM is modified to introduce higher-order approximations using MLS. The proposed method is validated in several steady and unsteady test cases. Numerical results show that the new method gives the formal order of accuracy, and it obtains more accurate results at lower mesh resolutions than standard second order methods.

- In Chapter 5 three high-order-preserving sliding-mesh methods based on Moving Least Squares approximants have been developed.

  In the simulation of rotationary geometries using a sliding mesh technique, the domain is discretized with different mesh regions. Since there is a relative movement between them, the intersection of the regions, denoted as interface, is not longer conformal. In order to couple the different regions and maintain the accuracy of the scheme, a high-order preserving methodology is required. Two distinct approaches were considered, namely the intersection-based and the *halo cell* sliding mesh methods. The latter is more flexible and easier to implement in any existing FV code, since it avoids the computation of intersections. As a drawback, it does not conserve mass from a theoretical point of view. However numerical results show that conservation errors are within the order of magnitude of the error committed on the variables.The accuracy and robustness of the new methodology has been investigated using various numerical

test for both inviscid compressible and viscous incompressible flows. Numerical results have shown that the proposed high-resolution sliding-mesh methods are able to preserve the formal order of accuracy of the high-order spatial discretization scheme. One clear advantage of the present methodology is that the same high-order discretization scheme is used through the whole computational domain involving both static and moving grids, naturally avoiding numerical artifacts. The capabilities of the developed formulation is tested with the simulation of a cross-flow turbine.

6.1 Further research

Although the high-order computational framework outlined in this dissertation provides by itself an accurate resolution, there are a number of areas in which additional capability could be added and the performance of the algorithm could be improved. Some of the directions that one can pursue to extend this research activity in the future are summarized below:

- Implement a numerical flux on the incompressible flow formulation.

  In Chapter 4, the convective flux was discretized with a deferred correction that combined the use of a first-order upwind with a centered high-order reconstruction. Despite the good results obtained in this dissertation, the use of numerical fluxes may be interesting for the simulation of turbulent flows.

- Extend the presented formulation to 3D unstructured meshes.

  Even tough a 3D numerical example has been presented in Chapter 3, a low-resolution structured mesh was employed. In this direction the complexity resides on the development of an efficient solver. We have a preliminary version of the code, but it has to be optimized. In order to achieve it, some of the following two directions must be taken into account.

- Implicit discretization for the density-based formulation.

  The current density-based FV-MLS framework developed in this thesis uses standard explicit time-marching schemes which have not been optimized for the high-order method. Several ways to improve the efficiency of the time integration procedure is to consider the use of multigrid and implicit algorithms. With the present formulation, the order of the FV-MLS method may be arbitrarily increased in stationary problems. However, for unsteady problems and explicit time-integration schemes, the maximum order achieved for the FV-MLS method is three.

- Multigrid techniques.

  The current pressure-based FV-MLS developed in this thesis uses an implicit time-marching schemes. Since an iterative procedure is used to impose incompressibility (SIMPLE algorithm), the systems of equations are solved each inner iteration until inner convergence is reached. In order to improve the efficiency of the numerical scheme, a multigrid method can be applied. Katz and Jameson in [102] proposed the use of a meshless technique for the development of a multigrid technique, denoted multicloud, achieving an interesting acceleration of convergence. Similar techniques could be used in this framework.

- Fluid-Structure Interaction (FSI).

  The capabilities of the sliding mesh technique on FSI are limited. The use body-fitted techniques obtains very accurate results in the flow simulation. However, when the solid is allowed to deform, there is a need to update the mesh surrounding, which is time consuming. In order to circumvent this difficulty, a different approach can be used, such as the Immersed Boundary (IB) method. The basic idea of a IB method is the use of a eulerian mesh to define the fluid and the solid is represented with a set of lagrangian points immersed on the fluid. In the IB method, there is not need to re-mesh and its efficiency has been proved. However, the resolution may not be as accurate as a body-fitted approach. In order to circumvent the accuracy problem, the use of the MLS approximations is proposed to transfer the information between the lagrangian and the eulerian description. A research line about this topic is already in progress.

- Numerical simulation of the cavitating flow.

  The appearance of the cavitating flow is a major cause of noise, vibration, erosion and efficacy loss in hydraulic machinery. Cavitation is the evaporation of a liquid in a flow when the pressure drops below the saturation pressure of that liquid. The importance of understanding cavitating flows is related to their occurrence in various technical applications, such as pumps, turbines and ship propellers, among others. In order to simulate this process, it is proposed to extend the presented density-based FV-MLS for all-speed to simulate the Navier-Stokes Korteweg equations, which simulates the phase transition phenomena [66]. A research line about this topic is already in progress.

Bibliography

[1] Abgrall, R., Marpen, F., *On Essentially Non-oscillatory Schemes on Unstructured Meshes: Analysis and Implementation*, Journal of Computational Physics, *114*(1):45–58, 1994.

[2] Anderson, J.D., *Computational Fluid Dynamics. The Basics with applications*, McGraw-Hill, 1995.

[3] Barth, T.J., Jespersen, D.C., *The design and application of upwind schemes on unstructured meshes*, AIAA-89-0366, 1989.

[4] Barth, T.J., *Recent developments in high order k-exact reconstruction on unstructured meshes*, AIAA paper, 93-0068, 1993.

[5] Bassi, F., Rebay, S., *A higher-order accurate discontinuous finite element solution of the 2D Euler equations*, Journal of Computational Physics *138*:251-285, 1997.

[6] Bassi, F., Rebay, S., *A higher-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations*, Journal of Computational Physics *131*:267-279, 1997.

[7] Bassi, F., De Bartolo, C., Hartmann, R., Nigro, A., *A discontinuous Galerkin method for inviscid low Mach number flows*, Journal of Computational Physics, *228*:3996-4011, 2009.

[8] Bazilevs, Y., Da Veiga, L.B., Cottrell, J.A., Hughes, T.J.R., Sangalli, G., *Isogeometric analysis: Approximation, stability and error estimates for h-refined meshes*, Mathematical Models and Methods in Applied Science, *16*:1031-1090, 2006.

[9] Bazilevs, Y., Calo, V.M., Cottrell, J.A., Hughes, T.J.R., Reali, A., Scovazzi, G., *Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows*, Computer Methods in Applied Mechanics and Engineering, *197*:173-201, 2007.

[10] Bazilevs, Y., Hsu, M.C., Akkerman, I., Wright, S., Takizawa, K., Henicke, B., Spielman, T., Tezduyar, T.E, *3D simulation of wind turbine rotors at full scale. Part I: Geometry modeling and aerodynamics*, International Journal of Numerical Methods in Fluids, 65:207–235, 2011.

[11] Bazilevs, Y., Korobenko, A., Deng, X., Yan, J., *Novel structural modeling and mesh moving techniques for advanced fluid-structure interaction simulation of wind turbines*, International Journal of Numerical Methods in Fluids, DOI: 10.1002/nme.4738, 2014.

[12] Belytschko, T., Krongauz,Y., Organ, D., Fleming, M., Krysl, P., *Meshless methods: An overview and recent developments*, Computer Methods in Applied Mechanics and Engineering , *139*: 3-47, 1996.

[13] Berger, M., LeVeque, R., *An adaptive cartesian mesh algorithm for the Euler equations in arbitrary geometries*, IAIAA Paper 89-1930-CP, 1989.

[14] Berger, M., Aftosmis, M.J., Murman, S.M., *Analysis of Slope Limiters on Irregular Grids*, AIAA Paper 2005-0490, 2005.

[15] Bilanceri, M., Beux, F., Salvetti, M.V., *Investigation on numerical schemes in the simulation of barotropic cavitating flows*, 7th International Symposium on Cavitation, CAV 2009. Paper 42, 2009.

[16] Blazer, J., *Computational Fluid Dynamics: Principles and Applications*, 2nd edition, Elsevier, 2005.

[17] Breviglieri, C., Azevedo, J.L., *High-order Implicit unstructured method for compressible flows*, Proceedings of COBEM 2011, 21th International Congress of Mechanical Engineering, Natal, Brazil, 2011.

[18] Brooks, A.N., Hughes, T.J.R., *Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations*, Computer Methods in Applied Mechanics and Engineering, *32*:199-259, 1982.

[19] Bujalsku, W., Jaworski, Z., Nienow, A.W., *ECFD Study of Homogenization with Dual Rushton Turbines—Comparison with Experimental Results: Part II: The Multiple Reference Frame*, Chemical Engineering Research and Design, *80*: 97-104, 2002.

[20] Caretto, L.S., Gosman, A.D., Patankar, S.V., Spalding, D.B., *Two calculation procedures for steady three dimensional flows with recirculation*, Proceeding of the 3rd International Conference on Numerical Methods in Fluid Dynamics , Paris (France), 1972.

[21] Charest, M.R.J., Groth, C.P.T., Gauthier, P.Q., *High-Order CENO Finite-Volume Scheme for Low-Speed Viscous Flows on Three-Dimensional Unstructured Mesh*, ICCFD7-1002 , 2012.

[22] Chassaing, J.C., Khelladi, S., Nogueira, X., *Accuracy assessment of a high-order moving least squares finite volume method for compressible flows*, Computer & Fluids 71:41-53, 2013.

[23] Cheng, Y.P., Lee, T.S., Low, H.T., Tao, W.Q., *Improvement of Simpler algorithm for Incompressible flow on Collocated grid system*, Numerical Heat Transfer 51:463-486, 2007.

[24] Chenoweth, S.K.M., Soria, J., Ooi, A., *A singularity-avoiding Moving-Least-Squares scheme for two-dimensional unstructured meshes*, Journal of Computational Physics, 228:5592–5619, 2009.

[25] Chiocchia, G., *Exact solutions to transonic and supersonic flows*, Technical Report AR-211, 1985.

[26] Choi, J., *Hybrid Spectral Difference/Embedded Finite Volume Method for Conservation Laws*, Submitted to Journal of Computational Physics, 2014.

[27] Choi, S.K., Yun, N.H., Cho, M., *Systematic Comparison of Finite Volume Methods with Staggered and Nonstaggered Grid Arrangements*, Numerical Heat Transfer 36:205-221, 1994.

[28] Choi, S.K., *Note on the Use of Momentum Interpolation Method of Unsteady Flows*, Numerical Heat Transfer 36:545-550, 1999.

[29] Chorin, A.J., *A numerical method for solving incompressible viscous flow problems*, Journal of Computational Physics, *2*: 12-26, 1967.

[30] Clarke, D., Salas, M., Hassan, H., *Euler calculations for multi-elements airfoils using Cartesian grids*, AIAA Journal, 24:353–358, 1986.

[31] Cockburn, B., Shu C.-W., *The local discontinuous Galerkin finite element method for convection-diffusion systems*, SIAM Journal on Numerical Analysis, *35*:2440-2463, 1998.

[32] Cockburn, B., Karniadakis, G., Shu C.-W., *Discontinuous Galerkin Methods. Theory, Computation and Applications.* , Lecture Notes in Computational Science and Engineering, Springer, 2000.

[33] Cockburn, B., Gopalakrishnan, J., Lazarov, R., *DUnified hybridization of discontinuous Galerkin, mixed and continuous Galerkin methods for second order elliptic problems* , SIAM Journal on Numerical Analysis, *47*:1319–1365, 2009.

[34] Colella, P., Woodward, P., *The piecewise parabolic method (PPM) for gas-dynamical simulations*, Journal of Computational Physics, *54*:174-201, 1984.

[35] Cordier, F., Degond, P., Kumbaro, A., *An Asymptotic-Preserving all-speed scheme for the Euler and Navier–Stokes equations*, Journal of Computational Physics, *231*(17):5685-5704, 2012.

[36] Cueto-Felgueroso, L., *Una visión general de los métodos sin malla: formulación y aplicaciones.*, Technical Report, Universidade da Coruña, 2002.

[37] Cueto-Felgueroso, L., *Partículas, Volúmenes Finitos y mallas no estructuradas: Simulación numérica de problémas de dinámica de fluidos*, PhD Thesis, Universidade da Coruña, 2005.

[38] Cueto-Felgueroso, L., Colominas, I., Fe, J., Navarrina, F., Casteleiro, M., *High order finite volume schemes on unstructured grids using Moving Least Squares reconstruction. Application to shallow waters dynamics*, International Journal for Numerical Methods in Engineering, *65*: 295-331, 2006.

[39] Cueto-Felgueroso, L., Colominas, I., Nogueira, X., Navarrina, F., Casteleiro, M., *Finite volume solvers and Moving Least-Squares approximations for the compressible Navier-Stokes equations on unstructured grids*, Computer Methods in Applied Mechanics and Engineering, *196*:4712-4736, 2007.

[40] Cueto-Felgueroso, L., Colominas, I., *High-order finite volume methods and multiresolution reproducing kernels*, Archives of Computational Methods in Engineering. *15*(2):185-228, 2008.

[41] Cueto-Felgueroso, L. and Peraire, J.,*A time-adaptive finite volume method for the Cahn–Hilliard and Kuramoto–Sivashinsky equations*, Journal of Computational Physics, 227(24): 9985-10017, 2008.

[42] Deconinck, H., Paillere, H., Struijs, R., Roe, P. L. *Multidimensional Upwind Schemes based on Fluctuation-Splitting for Systems of Conservation Laws*, Computational Mechanics, *11*:1083-1106, 2004.

[43] Dellacherie, S., *Analysis of Godunov type schemes applied to the compressible Euler system at low Mach number*, Journal of Computational Physics, *229*(4):978-1016, 2010.

[44] Demirdžić, I., Lilek, Ž., Peric, M., *A Collocated Finite Volume Method for predicting flows at all speeds*, International Journal for Numerical Methods in Fluids. *16*:1029-1050, 1993.

[45] Digraskar, D.A., *Simulations of Flow over Wind Turbines*, Master Thesis 1896. Paper 417, University of Massachusetts, 2014.

[46] Donea, J., *A Taylor-Galerkin method for convective transport problems*, International Journal for Numerical Methods in Engineering, *20*: 101-120, 1984.

[47] Donea, J., Huerta, A., *Finite Element Methods for Flow Problems*, Wiley, 2003.

[48] Dumbser, M., Titarev, V.A., Toro, E.F., *Quadrature-free non oscillatory finite volume schemes on unstructured meshes for non-linear hyperbolic systems*, Journal of Computational Physics, *221*: 693-723, 2007.

[49] Dumbser, M., Enaux, C., Toro, E.F., *Finite Volume Schemes of Very High Order of Accuracy for Stiff Hyperbolic Balance Laws*, Journal of Computational Physics, *227*: 3971-4001, 2008.

[50] Emeis, S., *Meteorlogical Explanation of Wake Clouds at Horns Rev Wind Farm*, Deutsches Widenergi Institut Magazin, 37:52–55, 2010.

[51] Erdos, J.I., Alznert, E., McNally, W., *Numerical solution of periodic transonic flow through a fan stage*, AIAA Journal, 15:1559–1568, 1977.

[52] Euler, L., *De Principiis Motus Fluidorum*, Novi Comm. Acad. Petrop., 15, 1-57, 1759.

[53] European Climate Foundation, *Roadmap 2050: a practical guide to a prosperous, low-carbon Europe*, `http://www.roadmap2050.eu/attachments/files/Roadmap2050-AllData-MinimalSize.pdf` (accessed February 25, 2015). 2010.

[54] Fadel, H., Agouzoul, M., Jimack, P.K., *High-order finite difference schemes for incompressible flows*, International Journal for Numerical Methods in Fluids. *00*:1-24, 2009.

[55] Feistauer, M., Kucera, V., *On a robust discontinuous Galerkin technique for the solution of compressible flow*, Journal of Computational Physics, *224*(1):208-221, 2007.

[56] Ferrer, E., Willden, R.H.J., *A High Order Discontinuous Galerkin Finite Element solver for the incompressible Navier-Stokes equations*, Computer & Fluids 46 (1):224-230, 2011.

[57] Ferrer, E.,*A high order discontinuous Galerkin–Fourier incompressible 3D Navier–Stokes solver with rotating sliding meshes for simulating cross-flow turbines*, PhD thesis, University of Oxford, 2012.

[58] Ferrer, E., Willden, R.H.J.,*A high order discontinuous Galerkin–Fourier incompressible 3d Navier–Stokes solver with rotating sliding meshes,* Journal of Computational Physics, 231:7037–7056, 2012.

[59] Ferziger, J.H., Peric, M., *Computational Methods for Fluid Dynamics*, 3rd edition Springer, 2002.

[60] Francois, B., Costes, M., Dufour, G.,*Comparison of Chimera and sliding mesh techniques for unsteady simulations of counter rotating open-rotors*, ISABE-2011-1231, 2011.

[61] Friedrich, O., *Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids*, Journal of Computational Physics, *144*:195-212, 1998.

[62] Gaitonde, D. V., Visbal, M. R., *High-Order Schemes for Navier-Stokes Equations: Algorithm and Implementation into FDL3DI*, U.S. Air Force Research Lab., TR AFRLVA-WP-TR-1998-3060, Wright-Patterson AFB, OH, 1998.

[63] Garcia, R., Bobenrieth, R.F., *Dettached Eddy simulation of the transonic flow over a circular cylinder*, Proceedings of COBEM 2005, 18th International Congress of Mechanical Engineering, Ouro Preto, Brazil, 2005.

[64] Ghia, U., Ghia, K.N., Shin, C.T., *High Re Solutions for Incompressible Flow Using the Navier Stokes Equations and a Multigrid Method*, Journal of Computational Physics, *48*:387-411, 1982.

[65] Godunov, S.K. *A Finite Difference Method for the Computation of Discontinuous Solutions of the Equations of Fluid Dynamics*, Mat. Sbornik, *47*:357-393, 1959.

[66] Gomez, H., Hughes, T.J.R., Nogueira, X., Calo, V.M., *Isogeometric analysis of the isothermal Navier–Stokes–Korteweg equations*, Computer Methods in Applied Mechanics and Engineering, *199*:1828-1840, 2010.

[67] Goude, A., *Fluid Mechanics of Vertical Axis Turbines: Simulations and Model Development*,Digital Comprehensive Summaries of Uppsala Dissertations from the Faculty of Science and Technology 998, 2012.

[68] Gourdain, N., Montagnac, M., Wlassow, F., Gazaix, M.,*High-performance computing to simulate large-scale industrial flows in multistage compressors*, International Journal of High Performance Computing Applications, 24:429–443, 2010.

[69] Guillard, H., Viozat, C., *On the behaviour of upwind schemes in the low mach number limit*, Computers and Fluids, *28*:63-86, 1999.

[70] Guillard, H., Murrone, A., *On the behaviour of upwind schemes in the low mach number limit: II. Godunov type schemes*, Computers and Fluids, *33*:655-675, 2004.

[71] Hardin, J. C., Ristorcelli, J. R., Tam, C. K. W., *ICASE/LaRC workshop on benchmark in computational aeroacoustic*, NASA Conference Publication 3300, 1995.

[72] Harlow, F., Welch, J., *Numerical Calculation of Time-dependent Viscous Incompressible Flow of Fluid with Free Surface*, Physics of Fluids, *8* (12):2182-2189, 1965.

[73] Harten, A., Osher, S., Engquist, B., Chakravarthy, S.,*Uniformly high order essentially non-oscillatory schemes, III*, Journal of Computational Physics, *71*:231-303, 1987.

[74] He, L.,*An euler solution for unsteady flows around oscillating blades*, Journal of Turbomachinery, 114:714–722, 1990.

[75] He, L.,*Method of simulating unsteady turbomachinery flows with multiple perturbations*, AIAA Journal, 30:2730–2735, 1992.

[76] Heinrich, T., Mallet, M., Mizukami, A., *An upwind finite element scheme for two-dimensional convective transport equation*, International Journal for Numerical Methods in Engineering, *11*: 131-144, 1986.

[77] Hoffman, J., Johnson, C., *Computability and adaptivity in CFD*, Encyclopedia of Computational Mechanics. , *3*, 2007.

[78] Hsin Wong, S.H., *Parallel Adaptive-Mesh Method for Predicting Flows Through Vertical-Axis Wind Turbines*, Master Thesis, University of Toronto, 2011.

[79] Hughes, T.J.R., Cottrell, J.A., Bazilevs, Y. *Isogeometric analysis: CAD, Finite elements, NURBS, exact geometry and mesh refinement*, Computer Methods in Applied Mechanics and Engineering, 194:4135-4195, 2005.

[80] Hughes, T.J.R., Franca, L.P., Hulbert,, G.M., *A new finite element formulation for computational fluid dynamics: I. Symmetric forms of the compressible Euler and Navier-Stokes equations and the second law of thermodynamics*, Computer Methods in Applied Mechanics and Engineering, *54*:223-234, 1986.

[81] Hughes, T.J.R., Franca, L.P., Hulbert,, G.M., *A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/Least-squares method for advective-diffusive equations*, Computer Methods in Applied Mechanics and Engineering, *73*:173-189, 1989.

[82] Hugoniot, H.,*Propagation des Mouvements dans les Corps er spécialement dans les Gaz Parfaits*, Journal de l'Ecole Polytechnique, *57*:3, 1887.

[83] Hussaini, M.Y., *On Large-Eddy Simulation of Compressible Flows*, AIAA paper, 98-2802, 1998.

[84] Iaccarino, G., *Turbulence modeling.* Lecture Notes ME469B, Stanford Junior University, 2004.

[85] IIT Knapur, *Lectures notes on Computational Fluid Dynamics and Heat Transfer*, `http://nptel.iitm.ac.in/courses/112104030/` (accessed October 15, 2013).

[86] Ijaz, M., Anand, N.K., *Co-Located Variables Approach Using Implicit Runge-Kutta Methods for Unsteady Incompressible Flow Simulation*, Numerical Heat Transfer B. *54*:291-313, 2008.

[87] Issa, R.I., *Solution of the Implicitly Discretised Fluid Flow Equations by Operator Splitting.* Journal of Computational Physics, *62*: 40-65, 1986.

[88] Ivan, L., Groth, C.P.T., *High-Order Central ENO Finite-Volume Scheme with Adaptive Mesh Refinement*, AIAA paper, 2007-4423, 2007.

[89] Janajreh, I., Qudaih, R., Talab, I., Ghenai, C.,*Aerodynamic flow simulation of wind turbine: Downwind versus upwind configuration*, Energy Conversion and Management, 51:1656–1663, 2010.

[90] Jansen, K.E., *A stabilized finite element method for computing turbulence*, Computer Methods in Applied Mechanics and Engineering, *174*, 299–317, 1999.

[91] Jaohindy, P., Ennamiri, H., Garde, F., Bastide, A., *Numerical investigation of airflow through a Savonius rotor*, Wind Energy, 17:853–868, 2014.

[92] Jaohindy, P., McTavish, S., Garde, F., Bastide, A., *An analysis of the transient forces acting on Savonius rotors with different aspect ratios*, Renewable Energy, 55:286–295, 2013.

[93] Jawahar, P., Kemath, H., *A high-resolution procedure for Euler and Navier-Stokes computations on unstructured grids*. Journal of Computational Physics, *164*: 165-203, 2000.

[94] Jiang, G.S., Shu, C.W., *Efficient Implementation of Weighted ENO schemes*, Journal of Computational Physics, *126*:202-228, 1996.

[95] Kacprzak, K., Liskiewicz, G., Sobczak, K., *Numerical investigation of conventional and modified Savonius wind turbines*, Renewable Energy, 60:578–585, 2013.

[96] Kampanis, N.A., Ekaterinaris, J.A., *A stagerred grid, high order accurate method for the incompressible Navier Stokes equations*. Journal of Computational Physics, *215*: 589-613, 2006.

[97] Kang, C., Iaccarino, G., Ham, F., *DNS of buoyancy-dominated turbulent flows on a bluff body using the immersed boundary method*, Journal of Computational Physics, *228*: 3189-3208, 2009.

[98] Kang, C., Chamorro, L., Hill, C., Arndt, R., Sotiropulos, F., *Flow structure interaction around an axial-flow hydrokinetic turbine: Experiments and CFD simulations*, Journal of Physics: Conference Series, *555* 012097, 2012.

[99] Kang, C., Liu, H., Yang, X., *Review of fluid dynamics aspects of Savonius-rotor-based vertical-axis wind rotors*, Renewable and Sustainable Energy Reviews, 33:499–508, 2014.

[100] Kang, Y.S., Sohn, D., Kim, J.H., Kim, H.G., Im, S., *A sliding mesh technique for the finite element simulation of fluid–solid interaction problems by using variable-node elements*, Computers and Fluids, 130:91–104, 2014.

[101] Karki, K., Patankar, S. V., *Pressure-based calculation procedure for viscous flows at all speeds in arbitrary configurations*, AIAA Journal, *27*(9):1167-1174, 1989.

[102] Katz, A., Jameson, A., *Multicloud: Multigrid convergence with a meshless operator*. Journal of Computational Physics, *228*: 5237-5250, 2009.

[103] Khelladi, S., Nogueira, X., Bakir, F., Colominas, I., *Toward a Higher-Order Unsteady Finite Volume Solver Based on Reproducing Kernel Particle Method*, Computer Methods in Applied Mechanics and Engineering, *200*: 2348–2362, 2011.

[104] Khosla, P.K., Rubin, S.G., *A diagonally dominant second-order accurate implicit scheme*, Computer & Fluids 2:207-209, 1974.

[105] Klainerman, S., Majda, A., *Compressible and incompressible fluids*, Communications on Pure and Applied Mathematics, *35*:629-651, 1982.

[106] Kopal, Z., *Tables of Supersonic Flow Around Cones*. Dept of Electrical Engineering, Center of Analysis, MIT, Cambridge (USA), 1947.

[107] Kovasznay, L.I.G., *Laminar flow behind a two dimensional grid*. Proc. Cambridge Philos. Soc. *44*: 58-62, 1948.

[108] Kwak, D., Chang, J.L.C., Shanks, S.P., Chakravarthy, S., *A three dimensional incompressible Navier Stokes Flow solver using primitive variables*. AIAA paper 84-0253, 1984.

[109] Kwak, D., Kiris, C.C., *Computation of Viscous Incompressible Flows*. Springer, 2011.

[110] Laizet, S., Lamballais, E.,*High order compact schemes for incompressible flows: A simple and efficient method with quasi spectral accuracy*. Journal of Computational Physics *228*,16: 5989–6015, 2009.

[111] Lancaster, P.,Salkauskas, K., *Surfaces generated by moving least squares methods*. Mathematics of Computation *37*,155: 141–158, 1981.

[112] Lancaster, P., Salkauskas, K., *Pseudo-divergence-free element free Galerkin method for incompressible fluid flow*, Computer Methods in Applied Mechanics and Engineering, 193:1119–1136, 2004.

[113] Lanzafame, R., Mauro, S., Messina, M., *2D CFD modeling of h-Darrieus wind turbines using a transition turbulence model*, Energy Procedia, 45:131–140, 2014.

[114] Lele, S. K., *Compact Finite Difference Schemes with Spectral-like Resolution*. Journal of Computational Physics *103*: 16-42, 1992.

[115] LeVeque, R. J., *Finite Volume Methods for Hyperbolic Problems*. Cambridge Texts in Applied Mathematics, Cambridge University Press *31*, 2002.

[116] Li, X.S, Gu, C.W. *An All-Speed Roe-type scheme and its asymptotic analysis of low Mach number behaviour*, Journal of Computational Physics, *227*:5144-5159, 2008.

[117] Li, X.S, Gu, C.W., *Mechamism of Roe-type schemes for all-speed flows and its application*, Computers and Fluids, *86*:56-70, 2013.

[118] Li, Y., Paik, K.J., Xing, T., Carrica, P.M., *Dynamic overset CFD simulations of wind turbine aerodynamics*, Renewable Energy, 37:285–298, 2012.

[119] Liepmann, H.W., Roshko, A.,*Elements of gasdynamics*, Dover, 2002.

[120] Liou, M.S., *A sequel to AUSM, Part II: AUSM$^+$-up for all speeds*, Journal of Computational Physics, *214*:137-170, 2006.

[121] Liu, G.R., Liu, M. B., *Smoothed Particle Hydrodynamics. A meshfree particle method.* World Scientific Publishing, Singapore: 2003.

[122] Liu, G.R., Gu, Y. T., *A point interpolation method.* Proc 4th Asia-Pacific Conference on Computational Mechanics, Singapore: 1009-1014, 1999.

[123] Liu, G.R., Gu, Y. T., Dai, K. Y., *Assessment and aplications of point interpolation methods for computational mechanics.* International Journal for Numerical methods in engineering, *59*: 1373-1397, 2004.

[124] Liu, Y., Vinokur, M., Wang, Z.J., *Spectral (finite) volume method for conservation laws on unstructured grids V: Extension to three-dimensional systems.* Journal of Computational Physics *212*: 454-472, 2006.

[125] Liu, W.K., Hao, W., Chen, Y., Jun, S., Gosz, J., *Multiresolution reproducing kernel particle methods.* Computational Mechanics, *20*: 295-309, 1997.

[126] Luo, J.Y., Issa, R.I., Gosman, A.D., *Prediction of impeller induced flows in mixing vessels using multiple frames of reference*, IChemE Symposium Series *136*: 549-556, 1994.

[127] Luo, H., Baum, J.D., Lohner, R., *A fast p-multigrid discontinuous Galerkin method for compressible flows at all speeds*, in: 44th AIAA Aerospace Sciences Meeting and Exhibit, Reeno, 2006.

[128] Lynch, P., *The Emergence of Numerical Weather Prediction: Ricahrdson's Dream.* Cambridge University Press, 2006.

[129] Majumdar, S., *Role of Underrelaxation in Momentum Interpolation for Calculation of Flow with Nonstaggered Grids*, Numerical Heat Transfer 13:125-132, 1988.

[130] Masatsuka, K., *I do like CFD, Vol.1*, Second edition, `http://www.cfdbooks.com/cfdbooks.html` (accessed February 25, 2015) , 2009.

[131] McNaughton, J., Afgan, I., Apsley, D.D., Rolfo, S., Stallard, T., Stansby, P.K., *A simple sliding-mesh interface procedure and its application to the CFD simulation of a tidal-stream turbine*, International Journal of Numerical Methods in Fluids,74:250–269, 2014.

[132] Merkle ,C. L. , Athavale, M., *Time Accurate Unsteady Incompressible Flow Algorithm Based on Artificial Compressibility*,AIAA 8th Computational Fluid Dynamics Conference, Honolulu, AIAA Paper 87-1137, 1987.

[133] Michalak ,K. , Ollivier-Gooch, C., *Limiters for Unstructured Higher-Order Accurate Solutions of the Euler Equations*, 46rd Aerospace Sciences Meeting and Exhibit, Reno, Nevada, AIAA Paper 2008-776, 2008.

[134] Miller, T.F., Schmidt, F.W., *Use of a Pressure-Weighted Interpolation Method for the Solution of Incompressible Navier-Stokes Equations on a Non-Staggered Grid System*, Numerical Heat Transfer 14:213-233, 1988.

[135] Minion, M.L., Brown, D.L., *Performance of Under resolved Two Dimensional Incompressible Flow Simlation, II*, Journal Computational Physics *138*:734-765, 1997.

[136] Mizukami, A., Hughes, T.J.R., *A Petrov-Galerkin finite element method for convection dominated flows: an accurate upwinding technique for satisfying the maximum principle*, Computer Methods in Applied Mechanics and Engineering *50*:181-193, 1985.

[137] Montlaur, A., Fernandez-Mendez, S., Huerta, A., *Discontinuous Galerking methods for the Stokes equations using divergence-free approximations*, International Journal for Numerical Methods in Fluids 57:1071-1092, 2008.

[138] Moon, Y.J., Cho, Y., Nam, H.S., *Computation of unsteady viscous flow and aeroacoustic noise of cross flow fans*, Computers and Fluids, 32:995–1015, 2003.

[139] Morinishi, Y., Lund, T.S., Vasilyev, O.V., Moin, P.,*Fully conservative higher order finite difference schemes for incompressible flow*, Journal Computational Physics *143*:90-124, 1998.

[140] Most, T., Bucher, C., *New concepts for moving least squuares: An interpolation non-singular weighting function and weighted nodal least square*, Engineering Analysis with Boundary Elements *32*:461-470, 2008.

[141] Munz, C. D., Roller, S., Klein, ,R., Geratz, K. J., *The extension of incompressible flow solvers to the weakly compressible regime*, Computers and Fluids, *32*(2):173-196, 2003.

[142] Murthy, V.S., Rose, W.C., *Detailed Measurements on a Circular Cylinder in Cross Flow*, AIAA Journal,*57*, 549–550, 1978.

[143] Navier, C.L.M.H., *Memoire sur les lois du mouvement des fluides*, Mem. Acad. Sci. Inst. France, 6, 389-440, 1822.

[144] Nayroles, B., Touzot, G., Villon, P., *Generalizing the finite element method: Diffuse approximation and diffuse elements*, Computational Mechanics *10* (5):307-318, 1992.

[145] Neeley, R., NASA research center, `https://www.nas.nasa.gov/Software/FAST/RND-93-010.walatka-clucas/htmldocs/chp_16.surferu.html` (accessed February 11, 2015).

[146] Nguyen, N.C., Peraire, J., Cockburn, B., *A Hybridizable Discontinuous Galerkin Method for the Incompressible Navier-Stokes Equations*, AIAA 2010-362, 2010.

[147] Nogueira, X., *Formulaciones de mínimos cuadrados móviles y de volúmenes finitos de alto orden para la simulacion numérica de flujos compresibles*, PhD Thesis, Universidade da Coruña, 2009.

[148] Nogueira, X., Cueto-Felgueroso, L., Colominas, I., Gómez, H., Navarrina, F., Casteleiro, M., *On the accuracy of Finite Volume and Discontinuous Galerkin discretizations for compressible flow on unstructured grids*, International Journal for Numerical Methods in Engineering, *78*:1553-1584, 2009.

[149] Nogueira, X., Cueto-Felgueroso, L., Colominas, I., Khelladi, S., *On the simulation of wave propagation with a higher order finite volume scheme based on Reproducing Kernel methods*, Computer Methods in Applied Mechanics and Engineering, *199*,155: 1471–1490, 2010.

[150] Nogueira, X., Cueto-Felgueroso, L., Colominas, I., Khelladi, S., Navarrina, F., Casteleiro, M. *Resolution of Computational Aeroacoustics problems on unstructured*

grids with a higher-order finite volume scheme, Journal of Computational and Applied Mathematics, *234*(7): 2089–2097, 2010.

[151] Nogueira, X., Cueto-Felgueroso, L., Colominas, I., Navarrina, F., Casteleiro, M., *A new shock-capturing technique based on Moving Least Squares for higher-order numerical schemes on unstructured grids*, Computer Methods in Applied Mechanics and Engineering,*199*: 2544-2558, 2010.

[152] Nogueira, X., Cueto-Felgueroso, L., Colominas, I., Gomez, H., *Implicit Large Eddy Simulation of non-wall-bounded turbulent flows based on the multiscale properties of a high-order finite volume method*, Computer Methods in Applied Mechanics and Engineering, *199*:615–624, 2010.

[153] Nogueira, X., Khelladi, S., Colominas, I., Cueto-Felgueroso, L., París, J., Gómez, H., *High-resolution finite volume methods on unstructured grids for turbulence and aeroacoustics*, Archives of Computational Methods in Engineering, *18*,3: 315–340, 2011.

[154] Nogueira, X., Taylor, B., Gómez, H., Colominas, I., Mackley, M. R.,*Experimental and computational modeling of oscillatory flow within a baffled tube containing periodic-tri-orifice baffle geometries*, Computers and Chemical Engineering,*49*: 1–17, 2013.

[155] Northwest Renewable News, *Hydrokinetic River Generator Gives Power To Remote Villages In Alaska*, `https://nwrenewablenews.wordpress.com/category/hydrokinetic/` (accessed February 25, 2015).

[156] Ollivier-Gooch, C.F., *High-order ENO schemes for unstructured meshes based on least squares reconstruction*, AIAA paper, 97-0540, 1997.

[157] Ollivier-Gooch, C.F., . *Quasi-ENO schemes for unstructured meshes based on unlimited data-dependent least-squares reconstruction*, Journal of Computational Physics, *133*:6-17, 1997.

[158] Ollivier-Gooch, C., Nejat, A., Michalak, K.,*Obtaining and verifying high-order unstructured finite volume solution to the Euler equations*, AIAA Journal, 47:2105–2120, 2009.

[159] Patankar, S. V.,*Numerical heat transfer and fluid flow*, McGraw-Hill, New York, 1980.

[160] Patankar, S.V.,*Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Corporation, Taylor and Francis Group, 1980.

[161] Peraire, J., Persson, P.O., *The Compact Discontinuous Galerkin (CDG) Method for Elliptic Problems*, SIAM Journal on Scientific Computing, *30*, N°4,1806-1824, 2008.

[162] Pereira, J.M.C., Kobayashi, M.H., Pereira, J.C.F., *A Fourth Order Accurate Finite Volume Compact Method for the Incompressible Navier Stokes Solutions*, Journal of Computational Physics. *167*:217-243, 2001.

[163] Peskin, C.S., *Flow patterns around heart valves: a numerical method*, Journal of Computational Physics, 10:252–271, 1972.

[164] Peyret, R., Taylor, T.D., *Computational Methods for Fluid Flow*, Springer Series in Computational Physics, Springer, 1983.

[165] PulseTidal, *Overview of Tidal Turbines*, `http://www.pulsetidal.co.uk/our-technology.html` (accessed February 11, 2015).

[166] Rai, M.,*A conservative treatment of zonal boundaries for Euler equation calculations*, Journal of Computational Physics, 62:472–503, 1986.

[167] Rai, M.,*A relaxation approach to patched-grid calculations with the Euler equations*, Journal of Computational Physics, 66:99–131, 1986.

[168] Rai, M.,*Navier-Stokes simulations of rotor-stator interaction using patched and overlaid grids*, Journal of Computational Physics, 3:387–396, 1987.

[169] Ramirez, L., *Nuevas Familias de Métodos de Volúmenes Finitos de alto toden basasdos en formulaciones Meshless para aplicaciones en Mecánica de Fluidos Computacional.*, Technical Report, Universidade da Coruña, 2012.

[170] Ramirez, L., Nogueira, X., Khelladi, S., Chassaing, J.C., Colominas, I., *A new higher-order finite volume method based on Moving Least Squares for the resolution of the incompressible Navier-Stokes equations on unstructured grids*, Computer Methods in Applied Mechanics and Engineering, *278*, 883–901, 2014.

[171] Ranjan, R., Pantano, C., *A collocated method for the incompressible Navier -Stokes equations inspired by the Box scheme*, Journal of Computational Physics, *232*:346-382, 2013.

[172] Rankine, W. J. M., *On the thermodynamic theory of waves of finite longitudinal disturbances*, Phil. Trans. Roy. Soc., *160*:277, 1870.

[173] Reed, W.H. and Hill, T.R., *T*riangular mesh methods for the neutron transport equation, Tech. Report LA-UR-73-479, Los Alamos Scientific Laboratory, Los Alamos,NM, (1973).

[174] Rhie, C.M., Chow, W.L., *Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation*, AIAA Journal 21:1525-1532, 1983.

[175] Richardson, L. F., *Weather Prediction by Numerical Process*, Cambidge University Press, 1922.

[176] Rieper, F., *On the Behaviour of Numerical Schemes in the Low Mach Number Regime*, PhD Thesis, Brandenburgischen Technischen Universität Cottbus, 2008.

[177] Rieper, F., Bader, G.,*The influence of cell geometry on the accuracy of upwind schemes in the low Mach number regime*, Journal of Computational Physics, *228*:2918-2933, 2009.

[178] Rieper, F., *On the dissipation mechanism of upwind-schemes in the low Mach number regime: A comparison between Roe and HLL*, Journal of Computational Physics, *229*:221-232, 2010.

[179] Rieper, F., *A low-Mach number fix for Roe's approximate Riemann solver*, Journal of Computational Physics, *230*:5263-5287, 2011.

[180] Ringleb, F., *Exakte Lösungen der Differentialgleichungen einer adiabatischen Gasströmung*, ZAMM Journal of Applied Mathematics and Mechanics, *20*:185-198, 1940.

[181] Roe, P. L., *Approximate Riemann solvers, parameter vectors and difference schemes*, Journal of Computational Physics,*43*, 357-372, 1981.

[182] Roller, S., Munz, C. D., *A low Mach number scheme based on multi-scale asymptotics*, Computing and Visualization in Science, *3*:85-91, 2000.

[183] Rusanov, V. V., *The calculation of the interaction of non-stationary shock waves and obstacles*, USSR Computational Mathematics and Mathematical Physics, *1*(2):304-320, 1962.

[184] Schäfer, M., Turek, S., *Benchmark Computations of Laminar Flow Around a Cylinder*, Flow Simulation with High-Performance Computers II, Volume 52 of Notes on Numerical Fluid Mechanics, Vieweg, pp. 547-566, 1996.

[185] Schmidt, S.J., Sezal, I.H., Schnerr, G. H., Talhamer, M., *Riemann techniques for the simulation of compressible liquid flow with phase-transition at all Mach numbers - Shock and wave dynamics in cavitating 3D Mmicro and macro systems*, AIAA paper 2008-1238,in: 46th AIAA Aerospace Sciences Meeting and Exhibit, Reeno, 2006.

[186] Schochet, S., *Fast singular limits of hyperbolic PDEs*, Journal of Differential Equations *114*:476-512, 1994.

[187] Sesterhenn, J., Müller, B., Thomann, H.,*On the cancellation problem in calculating compressible low Mach number flows*, Journal of Computational Physics, *151*(2):597-615, 1999.

[188] Sezai, I., *SIMPLE Method for the solution of incompressible flows on non-stagerred grids*, Eastern Mediterranean University internal Report, 2011.

[189] Shakib, F., Hughes, T.J.R., Johan, Z., *A new finite element formulation for computational fluid dynamics: X. The compressible Euler and Navier-Stokes equations*. Computer Methods in Applied Mechanics and Engineering, *89*, 141–219, 1991.

[190] Shima, E., Kitamura, K., *On New Simple Low-Dissipation Scheme of AUSM-Family for All Speeds*, *AIAA paper* 2009-136, 2009.

[191] Shima, E., Kitamura, K., *Performance of Low-Dissipation Euler Fluxes and Preconditioned Implicit Schemes in Low Speeds*, Communications in Computational Physics, *10*:90-119, 2011.

[192] Shu, C.W., Osher, S., *Efficient implementation of essentially nonoscillatory shock-capturing schemes*, Journal of Computational Physics, *77*, 439–471, 1998.

[193] Spyropoulos, E.T., Blaisdell, G.A., *Evaluation of the Dynamic Model for Simulations of Compressible Decaying Isotropic Turbulence*, *AIAA Journal 34*, 990-998 1996.

[194] Steijl, R., Barakos, G., *Sliding mesh algorithm for cfd analysis of helicopter rotor–fuselage aerodynamics*, International Journal of Numerical Methods in Fluids, 58:527–549, 2008.

[195] Steger, J., Dougherty, F., Benek, J., *A Chimera Grid Scheme*, ASME Mini-Symposium on Advances in Grid Generation, Houston, June 1982.

[196] Steltz, W.G., *Mechanical Engineers' Handbook: Energy and Power*, Wiley. Chapter 26, 2006.

[197] Stokes, G.G., *On the theories of the internal friction of fluids in motion*, Trans. Camb. Phil. Soc., 8, 287-305, 1845.

[198] Strategic Iniciative for Ocean Energy, *Wave and Tidal Energy Market Deployment Strategy for Europe*, `http://www.si-ocean.eu/en/upload/docs/SIOcean_Market_Deployment_Strategy-Web.pdf` (accessed February 25, 2015).

[199] Sun, L., Mathur, S., Murthy, Y., *An Unstructured Finite Volume Method for Incompressible Flows with Complex Immersed Boundaries*, Numerical Heat Transfer B. *58*:217-241, 2010.

[200] Tezduyar, T.E., Mittal, S., Ray, S.E., Shihm R., *Incompressible flow computations with stabilized bilinear and linear equal-order-interpolation velocity-pressure elements*, Computer Methods in Applied Mechanics and Engineering, *95*:221-242, 1992.

[201] Thom, A., *The Flow Past Circular Cylinders at Low Speeds*, Proc. Royal Society, *A141*:651-666, 1933.

[202] Thornber, B., *Implicit Large Eddy Simulation for Unsteady Multi-Component Compressible Turbulent flows*, PhD Thesis, Cranfield University, 2007.

[203] Thornber, B., Mosedale, A., Drikakis, D., Youngs, D., Williams, R.J.R., *An improved reconstruction method for compressible flows with low Mach number features*, Journal of Computational Physics, *227*:4873-4894, 2008.

[204] Toffolo, A., *On the theoretical link between design parameters and performance in cross-flow fans: a numerical and experimental study*, Computers and Fluids, 34:49–66, 2005.

[205] Toro, E.F., *Riemann Solvers and Numerical Methods for Fluid Dynamics. A Practical Introduction*, Second edition, Springer, 1999.

[206] Toro, E.F., Titarev, V.A., *Solution of the Generalised Riemann Problem for Advection-Reaction Equations*, Proc. Royal Society A, *458*:271-281, 2002.

[207] Trantin, P., *Isogeometric analysis of Euler compressible flow. Application to aerodynamics*, AIAA paper, 2012-0295, 2012.

[208] Turkel, E., *Preconditioned methods for solving the incompressible and low speed compressible equations*, Journal of Computational Physics, *72*:277-298, 1987.

[209] University of Malta, *A greener Malta*, THINK program `http://www.um.edu.mt/think/a-greener-malta/` (accessed February 25, 2015).

[210] Van Albada,G.D., Van Leer, B., Roberts, W.W., *A Comparative study of computational methods in cosmic gas dynamics*, Astronomy and Astrophysics, *108*, 76-84, 1982.

[211] Van Doormal, J.P., Raithby, G.B., *Enhancements of the SIMPLE Method for Predicting Incompressible Fluid Flows*, Numerical Heat Transfer 7:147-163, 1984.

[212] Van Houten, A., *Floating to the Future of Tidal Energy*, Marine Renewables Canada, Ottawa 2013 `http://www.marinerenewables.ca/wp-content/uploads/2013/11/AnneVanHouton.pdf`, (accessed February 25, 2015).

[213] Van Leer, B., *Towards the ultimate conservative difference scheme V. A second order sequel to Godunov's method*, Journal of Computational Physics, *32*:101, 1979.

[214] Van Leer, B., *Flux vector splitting for the Euler equations*, Lecture Notes in Physics, 170. Springer Verlag, 1982.

[215] Van Leer, B., Lee, W.T., Roe, P.L., *Characteristic time-stepping or local preconditioning of the Euler equations*,AIAA Computational Fluid Dynamics Conference, 10th, Honolulu, HI, June 24-27, 1991, Technical Papers (A91-40701 17-34). Washington, DC, American Institute of Aeronautics and Astronautics, 260-282, 1991.

[216] Van Rosendale, J., *Floating shock fitting via Lagrangian adaptive meshes*, ICASE 94-89, 1989.

[217] Venkatakrishnan, V., *Convergence to steady state solutions of the Euler equations on unstructured grids with limiters*, Journal of Computational Physics, *118*, 120-130, 1995.

[218] V. Venkatakrishnan, *On the accuracy of limiters and convergence to steady state solutions*, 31st Aerospace Sciences Meeting and Exhibit, Reno, Nevada, AIAA Paper 93-0880, 1993.

[219] Venkateswaran, S., Merkle, C. L., *Dual time stepping and preconditioning for unsteady computations*, 33rd Aerospace Sciences Meeting and Exhibit, Reno, Nevada, AIAA Paper 95-0078, 1995.

[220] Versteeg, H.K., Malalasekera, W., *An Introduction to Computational Fluid Dynamics. The Finite VOlume Method*, 2nd edition Pearson Prentice Hall, 2007.

[221] Volker, J., Gunar, M., *Higher order finite element discretization in a benchmark problem for incompressible flows*, International Journal for Numerical Methods in Fluids 37:885-903, 2001.

[222] Wang, X., *CFD Simulation of Complex Flows in Turbomachinery and Robust Optimization of Blade Design*, PhD Thesis, Vrije Universiteit Brussel, 2010.

[223] Wang, Z.J., Hariharan, N., Chen, R., *Recent development on the conservation property of Chimera*, International Journal of Computational Fluid Dynamics, 15:265–278, 2001.

[224] Wang, Z.J., *Spectral (finite) volume method for conservation laws on unstructured grids*, Journal of Computational Physics *178*:210-251, 2002.

[225] Wendt, J.F., *Computational Fluid Dynamics: an Introduction*, First edition, Springer, 1992.

[226] Williamson, C.H.K., *Vortex dynamics in the cylinder wake*, Annual Reviews Fluid Mechanics *28*:477-539, 1996.

[227] Xu, C.Y., Zhao, L.Q., Sun, J.H., *Large-eddy simulation of the compressible flow past a tabbed cylinder*, Chinese Science Bulletin, *57*, 3203–3210, 2012.

[228] Ye, T., Mittal, R., Udaykumar, H.S., Shyy, W., *An accurate cartesian grid method for viscous incompressible flows with complex immersed boundaries*, Journal of Computational Physics, 156:209–249, 1999.

[229] Yousefifard, M., Ghadimi, P., Zamanian, R., *Unstructured Grid Solutions for Incompressible Laminar Flow over a Circular Cylinder Using Particular Finite Volume-Finite Element Method*, Journal of Engineering, 2013, ID 795237, 2013.

[230] Yu, B., Tao, W., Wei, J., *Discussion on Momentum Interpolation Method for Collocated Grids of Incompressible Flow*, Numerical Heat Transfer 42:141-166, 2002.

[231] Zanhle, F., Sørensen, N., *Overset Grid Flow Simulation on a Modern Wind Turbine*, AIAA paper 2008-6727, 2008.

[232] Zhang, N., He, A., Ballard, B., Saxon, R., Schippers, W., *Simulation of the Motion of a Hydro Turbine Using the Immersed-Boundary Method*, AIAA paper 2012-1234,in: 50th AIAA Aerospace Sciences Meeting , 2012.

[233] Zhang, S., Zhao, X., Bayyuk, S., *Generalized formulations for the Rhie-Chow interpolation*, Journal of Computational Physics 258:880-914, 2014.

[234] Zienkiewicz, O.C., Cheung, Y.K., *Finite elements in the solution of field problems*, The Engineer, 507-510, 1965.